Informatica® Cloud Application Integration
November 2023

Salesforce and Application Integration

Informatica Cloud Application Integration Salesforce and Application Integration
November 2023

Publication Date: 2023-11-07

Table of Contents

Preface

Use *Salesforce and Application Integration* to learn how to install the Salesforce Managed Package and run a guide within a Salesforce organization.

CHAPTER 1

Salesforce Managed Package

This chapter includes the following topics:

Salesforce Managed Package Overview

You can use the Informatica Cloud Real Time for Salesforce package to:

- Integrate Application Integration with Salesforce.
- Test and publish a Salesforce connection in Application Integration to fetch the metadata from Salesforce.
- Work with Salesforce objects when you run a process.
- Work with Salesforce guides.
- Access Application Integration and Salesforce resources with a single sign-on.
- Use a Salesforce outbound message (OBM) that triggers an Application Integration process. For more information, see the *Salesforce Outbound Messages* topic in the *System Services, Listeners and Connectors* section of the *Design* guide.

Salesforce managed package URLs

The Salesforce managed package for Informatica Cloud Real Time version 1.26 uses version 56.0 of the Salesforce API.

You can install or upgrade to the latest managed package version 1.26.

Log in to the following URLs to install or upgrade to the latest managed package:

- To install the latest version 1.26, use the following URL: https://login.salesforce.com/packaging/installPackage.apexp?p0=04t5c000000o7Tk

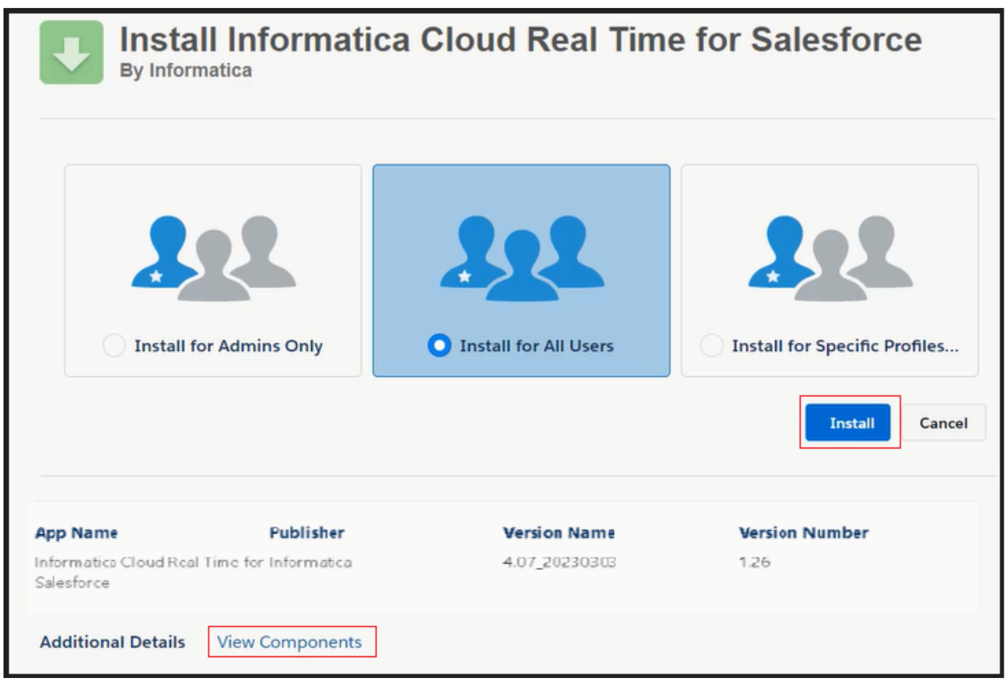
- To install the latest sandbox version, use the following URL:
https://test.salesforce.com/packaging/installPackage.apexp?p0=04t5c000000o7Tk

Installing the Salesforce Managed Package

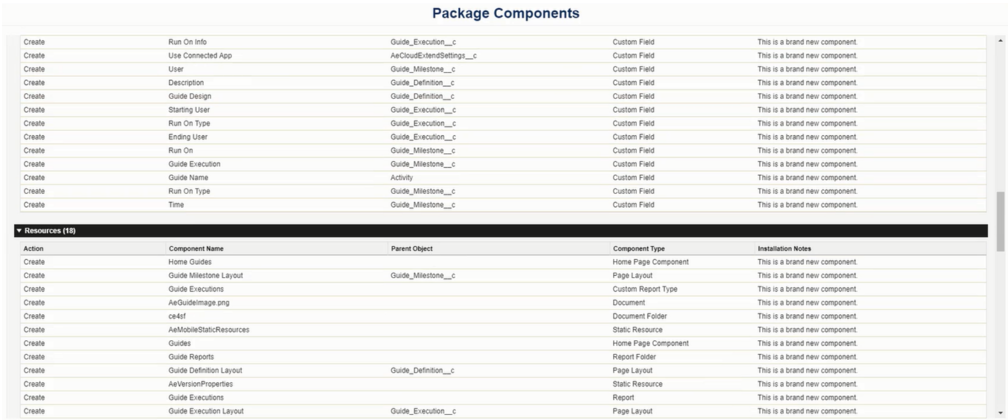
Before you install the Salesforce managed package, make sure that your organization meets the following prerequisites:

- Your organization uses the Unlimited, Enterprise, Professional, or Developer edition of Salesforce. However, when you use the trial version which is the Professional edition of Salesforce, you might not be able to configure the managed package or connect to Salesforce. This happens because the API Enabled option is not available for the trial organizations. Informatica recommends that you use the Unlimited, Enterprise, or Developer edition of Salesforce.
- Salesforce APIs are enabled.
- You have administrative rights in the Salesforce organization where you want to install the managed package.

1. Depending on your business need, log in to your production or sandbox Salesforce developer account.
   - Production: https://login.salesforce.com/packaging/installPackage.apexp?p0=04t5c000000o7Tk
   - Sandbox: https://test.salesforce.com/packaging/installPackage.apexp?p0=04t5c000000o7Tk
2. Enter your Salesforce developer account user credentials.
   On the Install **Informatica Cloud Real Time for Salesforce** page, select one of the following options:
   - Install for Admins Only. Select this option to allow only admin users in Salesforce to run guides.
   - Install for All Users. Select this option to allow all users in Salesforce to run guides on the objects they work with. Most organizations grant access to all users.
   - Install for Specific Profiles. Select this option to configure custom security settings.



3. Optionally, click **View Components** to see detailed components of the managed package.



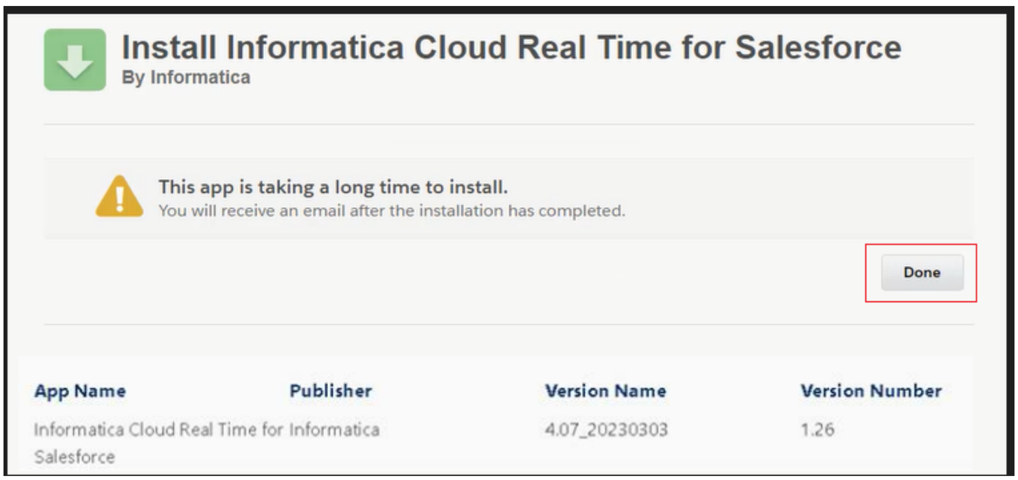
Package Components

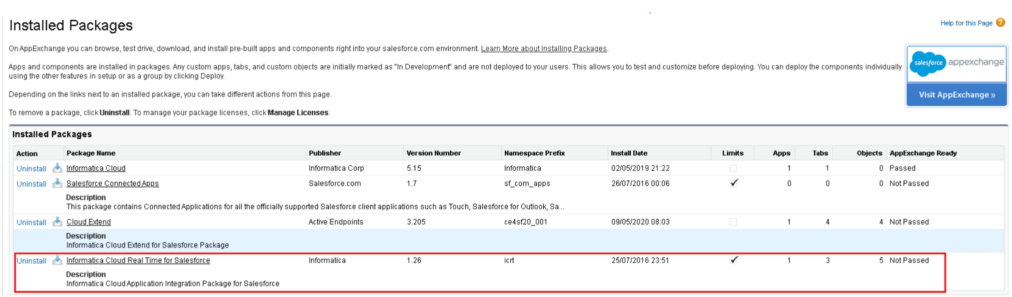
Create	Run On Info	Guide_Execution__c	Custom Field	This is a brand new component.
Create	Use Connected App	AeCloudExtendSettings__c	Custom Field	This is a brand new component.
Create	User	Guide_Milestone__c	Custom Field	This is a brand new component.
Create	Description	Guide_Definition__c	Custom Field	This is a brand new component.
Create	Guide Design	Guide_Definition__c	Custom Field	This is a brand new component.
Create	Starting User	Guide_Execution__c	Custom Field	This is a brand new component.
Create	Run On Type	Guide_Execution__c	Custom Field	This is a brand new component.
Create	Ending User	Guide_Execution__c	Custom Field	This is a brand new component.
Create	Run On	Guide_Milestone__c	Custom Field	This is a brand new component.
Create	Guide Execution	Guide_Milestone__c	Custom Field	This is a brand new component.
Create	Guide Name	Activity	Custom Field	This is a brand new component.
Create	Run On Type	Guide_Milestone__c	Custom Field	This is a brand new component.
Create	Time	Guide_Milestone__c	Custom Field	This is a brand new component.

▾ Resources (18)

Action	Component Name	Parent Object	Component Type	Installation Notes
Create	Home Guides		Home Page Component	This is a brand new component.
Create	Guide Milestone Layout	Guide_Milestone__c	Page Layout	This is a brand new component.
Create	Guide Executions		Custom Report Type	This is a brand new component.
Create	AeGuideImage.png		Document	This is a brand new component.
Create	ce4sf		Document Folder	This is a brand new component.
Create	AeMobileStaticResources		Static Resource	This is a brand new component.
Create	Guides		Home Page Component	This is a brand new component.
Create	Guide Reports		Report Folder	This is a brand new component.
Create	Guide Definition Layout	Guide_Definition__c	Page Layout	This is a brand new component.
Create	AeVersionProperties		Static Resource	This is a brand new component.
Create	Guide Executions		Report	This is a brand new component.
Create	Guide Execution Layout	Guide_Execution__c	Page Layout	This is a brand new component.

4. Click **Install**.
5. Click **Done** after the install is complete.
   If the installation takes a long time, a message appears stating that you will receive an email when the installation is complete. Click **Done**.





Installed Packages

Help for this Page

On AppExchange you can browse, test drive, download, and install pre-built apps and components right into your salesforce.com environment. Learn More about Installing Packages.

Apps and components are installed in packages. Any custom apps, tabs, and custom objects are initially marked as "In Development" and are not deployed to your users. This allows you to test and customize before deploying. You can deploy the components individually using the other features in setup or as a group by clicking Deploy.

Depending on the links next to an installed package, you can take different actions from this page.

To remove a package, click **Uninstall**. To manage your package licenses, click **Manage Licenses**.

salesforce appexchange
Visit AppExchange »

Installed Packages

Action	Package Name	Publisher	Version Number	Namespace Prefix	Install Date	Limits	Apps	Tabs	Objects	AppExchange Ready
Uninstall	Informatica Cloud	Informatica Corp	5.15	Informatica	02/05/2019 21:22		1	1	0	Passed
Uninstall	Salesforce Connected Apps	Salesforce.com	1.7	sf_com_apps	26/07/2016 00:06	✓	0	0	0	Not Passed
	Description This package contains Connected Applications for all the officially supported Salesforce client applications such as Touch, Salesforce for Outlook, Sa...									
Uninstall	Cloud Extend	Active Endpoints	3.205	ce4sf20_001	09/05/2020 08:03		1	4	4	Not Passed
	Description Informatica Cloud Extend for Salesforce Package									
Uninstall	Informatica Cloud Real Time for Salesforce	Informatica	1.26	icrt	25/07/2016 23:51	✓	1	3	5	Not Passed
	Description Informatica Cloud Application Integration Package for Salesforce									

The **Installed Packages** page appears with the managed package under **Installed Packages**.

6. Click **All Tabs** at the top of the page.

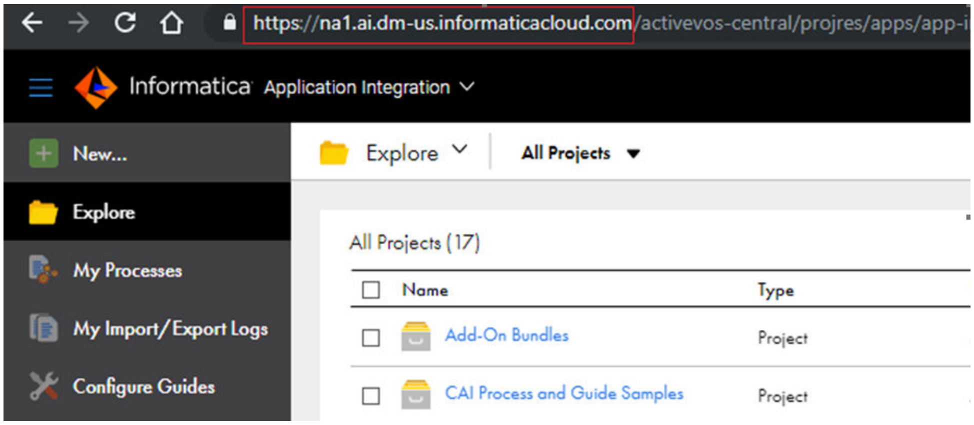
7. On the **All Tabs** page, click **Guide Setup**.

8. On the **Registration Verification** page, enter the Informatica Cloud Real Time Host URL.

   To construct the Informatica Cloud Real Time Host URL, you will need the following values:

   Cloud Application Integration URL

   Log in to Informatica Intelligent Cloud Services℠ and select the **Application Integration** service. The URL portion that is highlighted in the following image is your Cloud Application Integration URL:



   POD-specific Cloud Application URL

   Use the POD-specific Cloud Application URL for Salesforce instead of the Cloud Application Integration URL. Otherwise, when you access Application Integration and the Salesforce organization linked to Application Integration simultaneously on two different browser tabs, you might encounter issues while publishing Application Integration assets.

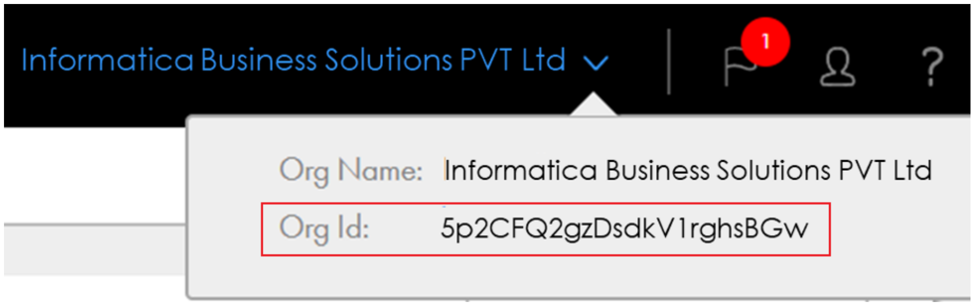
   Determine your POD-specific Cloud Application URL for Salesforce based on your POD and Cloud Application Integration URL. The following table lists the POD-specific Cloud Application URL for Salesforce that is applicable for various PODs and Cloud Application Integration URLs:

POD Name	Cloud Application Integration URL	POD-specific Cloud Application URL for Salesforce
AP NorthEast 2	`https://apne2-cai.dm-apne.informaticacloud.com`	`https://apne2-sfdc-cai.dm-apne.informaticacloud.com`
APJ/APSE1	`https://apse1-cai.dm-ap.informaticacloud.com`	`https://apse1-sfdc-cai.dm-ap.informaticacloud.com`
AZ GERMANY	`https://emc1-cai.dm1-em.informaticacloud.com`	`https://emc1-sfdc-cai.dm1-em.informaticacloud.com`
AZ SINGAPORE	`https://apse1-cai.dm1-apse.informaticacloud.com`	`https://apse1-sfdc-cai.dm1-apse.informaticacloud.com`
AZURE NTT	`https://apne1-cai.dm1-ap.informaticacloud.com`	`https://apne1-sfdc-cai.dm1-ap.informaticacloud.com`
AZURE POD1	`https://usw1-cai.dm1-us.informaticacloud.com`	`https://usw1-sfdc-cai.dm1-us.informaticacloud.com`

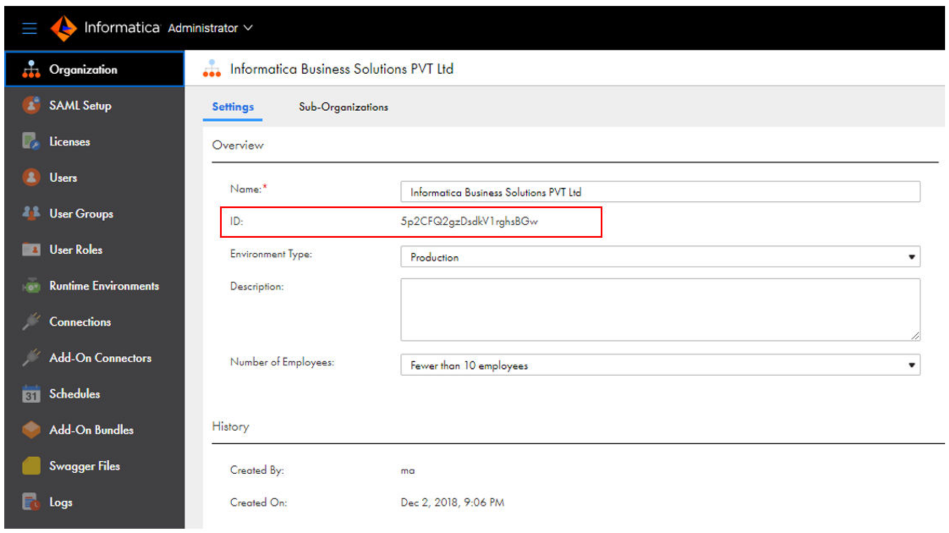
POD Name	Cloud Application Integration URL	POD-specific Cloud Application URL for Salesforce
AZURE POD3	https://usw3-cai.dm1-us.informaticacloud.com	https://usw3-sfdc-cai.dm1-us.informaticacloud.com
C360-Preview	https://usw1-cai.dmp-us.informaticacloud.com	https://usw1-sfdc-cai.dmp-us.informaticacloud.com
CANADA	https://nac1-cai.dm-na.informaticacloud.com	https://nac1-sfdc-cai.dm-na.informaticacloud.com
EMW1	https://emw1-cai.dm-em.informaticacloud.com	https://emw1-sfdc-cai.dm-em.informaticacloud.com
GCP	https://usw1-cai.dm2-us.informaticacloud.com	https://usw1-sfdc-cai.dm2-us.informaticacloud.com
POD1/ USW1/NA1	https://na1.ai.dm-us.informaticacloud.com	https://na1.sfdc-ai.dm-us.informaticacloud.com
POD2/NA2	https://na2.ai.dm-us.informaticacloud.com	https://na2.sfdc-ai.dm-us.informaticacloud.com
POD3/USW3	https://usw3-cai.dm-us.informaticacloud.com	https://usw3-sfdc-cai.dm-us.informaticacloud.com
POD4/USE4	https://use4-cai.dm-us.informaticacloud.com	https://use4-sfdc-cai.dm-us.informaticacloud.com
POD5/USW5	https://usw5-cai.dm-us.informaticacloud.com	https://usw5-sfdc-cai.dm-us.informaticacloud.com
POD6/USE6	https://use6-cai.dm-us.informaticacloud.com	https://use6-sfdc-cai.dm-us.informaticacloud.com
PREVIEW	https://usw1-cai.dmr-us.informaticacloud.com	https://usw1-sfdc-cai.dmr-us.informaticacloud.com
UAE (Azure EMSE)	https://emse1-cai.dm1-emse.informaticacloud.com	https://emse1-sfdc-cai.dm1-emse.informaticacloud.com
UK1	https://uk1-cai.dm-uk.informaticacloud.com	https://uk1-sfdc-cai.dm-uk.informaticacloud.com

Organization ID

To view your Informatica Organization ID, log in to Informatica Intelligent Cloud Services and copy the Organization ID from **Organization** list shown in the following image:



Alternatively, you can select the **Administrator** service and then click **Organization**. Copy the Informatica Organization ID displayed in the **ID** field as shown in the following image:

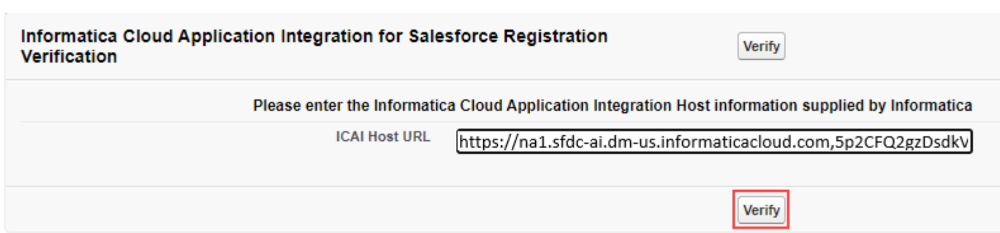


Construct the Informatica Cloud Application Integration Host URL by using the following format:

`<POD-specific Cloud Application URL for Salesforce>,<Informatica Organization ID>`

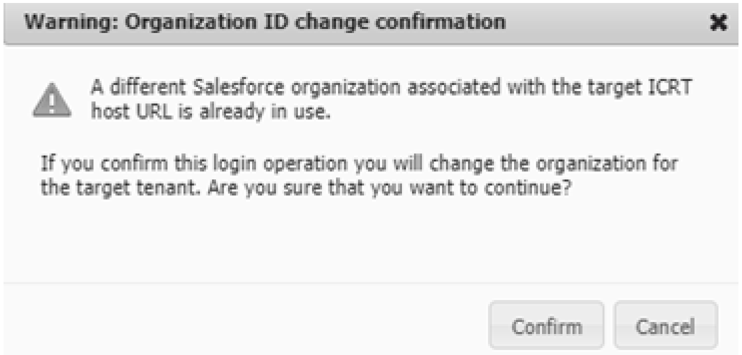
Enter the URL in the **ICAI Host URL** field.

For example, enter `https://na1.sfdc-ai.dm-us.informaticacloud.com,5p2CFQ2gzDsdkV1rghsBGw` as shown in the following image:



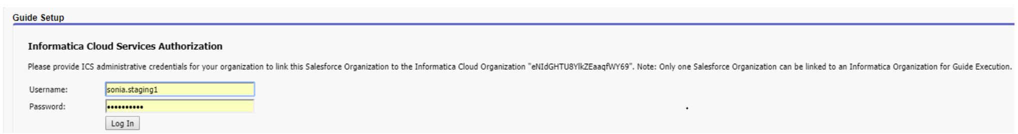
9. Click **Verify**.
   The **Informatica Cloud Services Authorization** page appears.

Note: If your Informatica Intelligent Cloud Services organization was previously linked to another Salesforce organization, a warning message appears as shown in the following image:

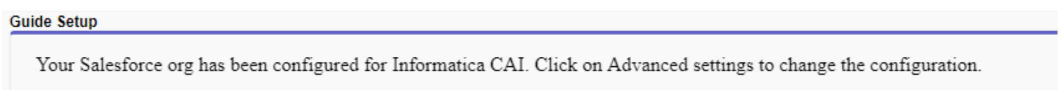


You can change the linked Salesforce organization when you install a managed package. For example, you have completed steps 1 through 8 to install the managed package in Salesforce organization A. However, your Informatica Intelligent Cloud Services organization B is already linked to a Salesforce organization C. If you click **Confirm** on the warning message, you link Informatica Intelligent Cloud Services organization B to Salesforce organisation A.

10. Enter your Informatica Intelligent Cloud Services administrator credentials and then click **Log In**.



A confirmation message appears as shown in the following image stating that your Salesforce organization has been configured:

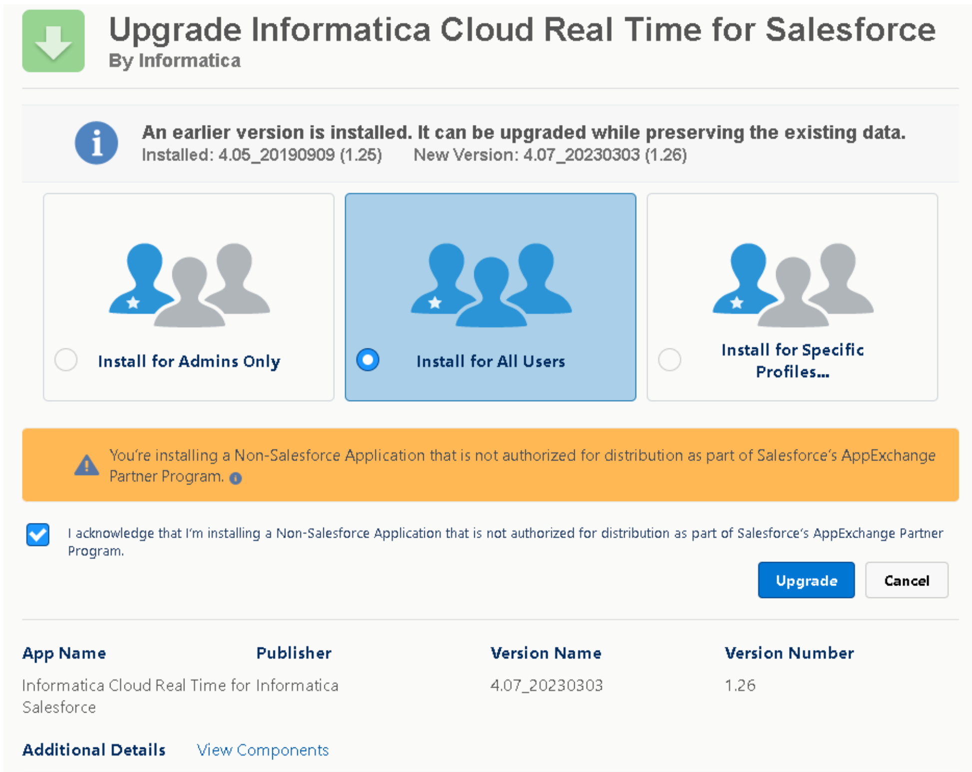


Upgrading the Salesforce Managed Package

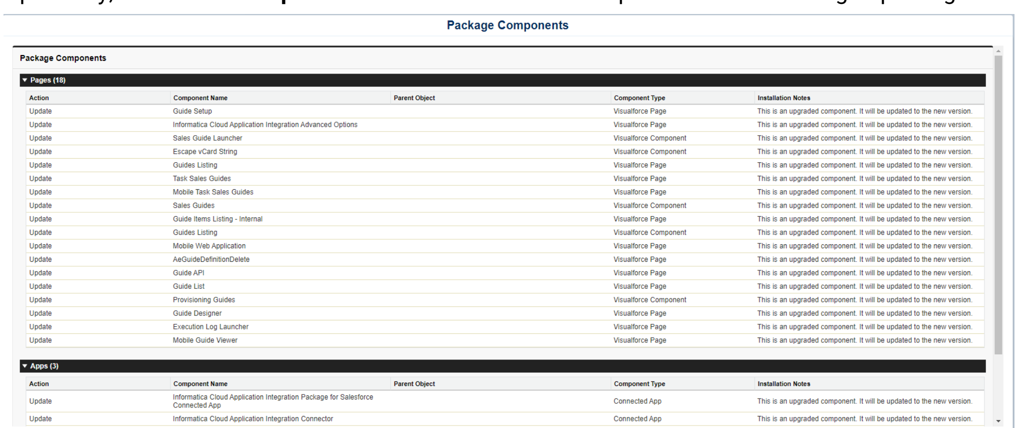
Perform the following steps to upgrade the Salesforce managed package:

1. Depending on your business need, log in to your production or sandbox Salesforce developer account.
   - Production: https://login.salesforce.com/packaging/installPackage.apexp?p0=04t5c000000o7Tk
   - Sandbox: https://test.salesforce.com/packaging/installPackage.apexp?p0=04t5c000000o7Tk
2. Enter your Salesforce developer account user credentials.
   On the **Upgrade Informatica Cloud Real Time for Salesforce** page, select one of the following options:
   - Install for Admins Only. Select this option to allow only admin users in Salesforce to run guides.
   - Install for All Users. Select this option to allow all users in Salesforce to run guides on the objects they work with. Most organizations grant access to all users.

- Install for Specific Profiles. Select this option to configure custom security settings.



3. Optionally, click **View Components** to see the detailed components of the managed package.



Package Components

Package Components

Pages (18)

Action	Component Name	Parent Object	Component Type	Installation Notes
Update	Guide Setup		Visualforce Page	This is an upgraded component. It will be updated to the new version.
Update	Informatica Cloud Application Integration Advanced Options		Visualforce Page	This is an upgraded component. It will be updated to the new version.
Update	Sales Guide Launcher		Visualforce Component	This is an upgraded component. It will be updated to the new version.
Update	Escape vCard String		Visualforce Component	This is an upgraded component. It will be updated to the new version.
Update	Guides Listing		Visualforce Page	This is an upgraded component. It will be updated to the new version.
Update	Task Sales Guides		Visualforce Page	This is an upgraded component. It will be updated to the new version.
Update	Mobile Task Sales Guides		Visualforce Page	This is an upgraded component. It will be updated to the new version.
Update	Sales Guides		Visualforce Component	This is an upgraded component. It will be updated to the new version.
Update	Guide Items Listing - Internal		Visualforce Page	This is an upgraded component. It will be updated to the new version.
Update	Guides Listing		Visualforce Component	This is an upgraded component. It will be updated to the new version.
Update	Mobile Web Application		Visualforce Page	This is an upgraded component. It will be updated to the new version.
Update	AeGuideDefinitionDelete		Visualforce Page	This is an upgraded component. It will be updated to the new version.
Update	Guide API		Visualforce Page	This is an upgraded component. It will be updated to the new version.
Update	Guide List		Visualforce Page	This is an upgraded component. It will be updated to the new version.
Update	Provisioning Guides		Visualforce Component	This is an upgraded component. It will be updated to the new version.
Update	Guide Designer		Visualforce Page	This is an upgraded component. It will be updated to the new version.
Update	Execution Log Launcher		Visualforce Page	This is an upgraded component. It will be updated to the new version.
Update	Mobile Guide Viewer		Visualforce Page	This is an upgraded component. It will be updated to the new version.

Apps (3)

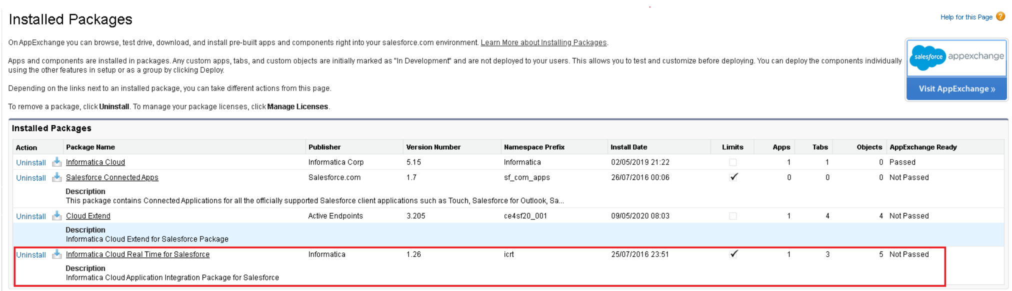
Action	Component Name	Parent Object	Component Type	Installation Notes
Update	Informatica Cloud Application Integration Package for Salesforce Connected App		Connected App	This is an upgraded component. It will be updated to the new version.
Update	Informatica Cloud Application Integration Connector		Connected App	This is an upgraded component. It will be updated to the new version.

4. Click **Upgrade**.

5. Click **Done** after the upgrade is complete.

Note: If the upgrade takes a long time, a message appears stating that you will receive an email when the upgrade is complete. Click **Done**.

The **Installed Packages** page appears with the upgraded managed package under **Installed Packages**.

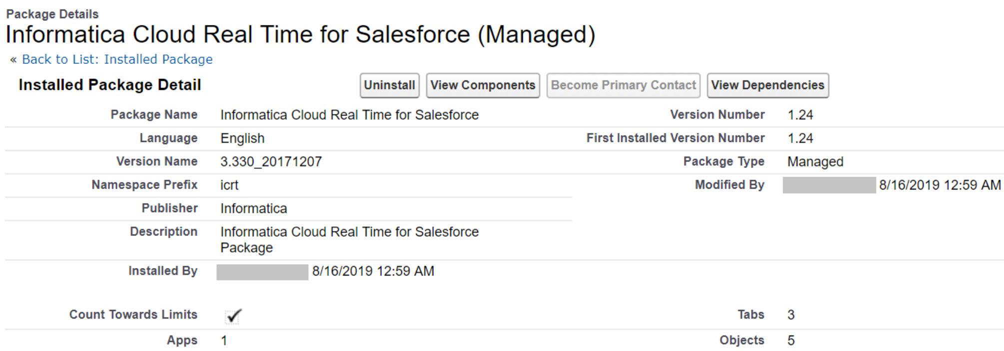


Uninstalling and Deleting the Salesforce Managed Package

The following steps detail the procedure to uninstall and delete the Salesforce managed package:

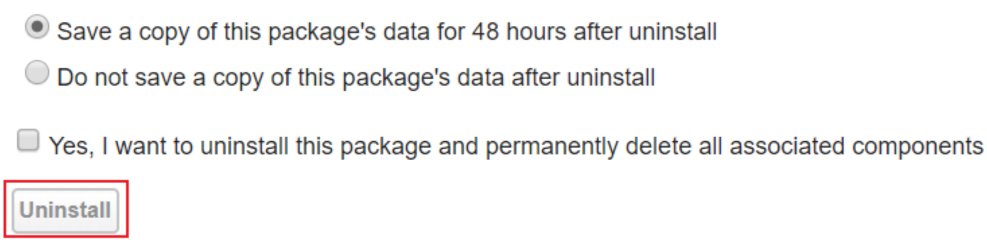
1. Log in to your Salesforce developer account.
2. Click **Setup**.
3. In the **Quick Find / Search** box, search for `Installed Packages`.
4. In the **Installed Packages** page, click **Informatica Cloud Real Time for Salesforce**.

   Note: Before you uninstall the managed package, ensure that you delete all the artifacts that depend on the Informatica Cloud Real Time for Salesforce. For example, custom pages, buttons, page layouts, Visualforce pages.
5. Click **Uninstall** at the top of the page.

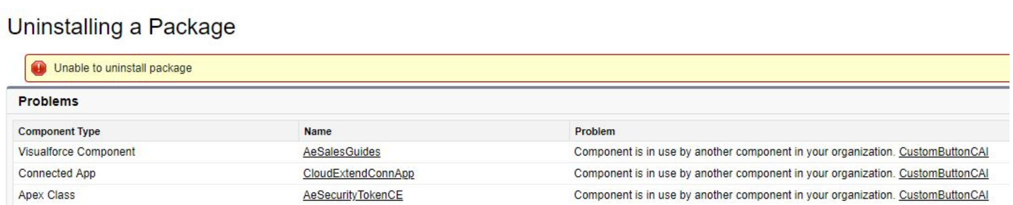


The **Uninstalling a Package** page appears with a list of all package components.

6. You can choose to save the package data for 48 hours after you uninstall the package. Click **Uninstall**.



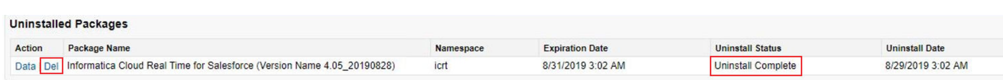
If you do not delete all the artifacts that depend on the Informatica Cloud Real Time for Salesforce, the uninstallation might fail.



Delete the dependent artifacts and uninstall the package again.

7. After you uninstall the package, the **Informatica Cloud Real Time for Salesforce** package appears under **Uninstalled Packages** on the **Installed Packages** page and the status displays as **In Progress**.

After the uninstallation is complete, the status displays as **Uninstall Complete**.



8. To delete the managed package, click **Del** under **Action**.
A dialog box prompts you to confirm if you want to delete the managed package. Click **OK**.

More Information

To learn more, see the *Using Salesforce System Services, Listeners and Connectors* section under *Design*.

CHAPTER 2

Running a Guide within a Salesforce Organization

This chapter includes the following topics:

Running a Guide within a Salesforce Organization Overview

To run a guide from within your Salesforce organization, create a guide in Application Integration and select a Salesforce object in the **Start** > **Applies To** property. You can open guides in Community Sites that are enabled with Salesforce enhanced domain.

Before you run a guide in Salesforce, you need to perform some tasks to prepare your organization.

Note: If you have migrated to the Informatica Intelligent Cloud Services℠, log in to Salesforce and verify that your guides are visible on the relevant Salesforce object pages. If you do not see your guides, log out of Salesforce, clear the browser cache, and then log in to Salesforce.

Step 1: Install the Salesforce Managed Package

Log in to Salesforce and install the Informatica Cloud Real Time for Salesforce package. The managed package connects your Informatica Intelligent Cloud Services and Salesforce accounts.

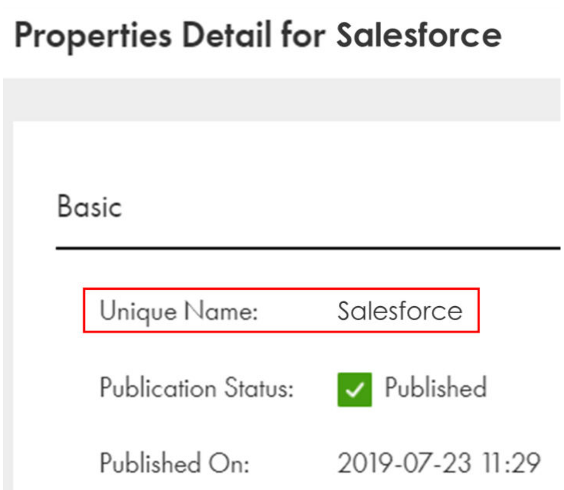
For information on how to install the managed package, see "Installing the Salesforce Managed Package" on page 7.

Step 2: Create a Salesforce Connection

Log in to Application Integration, and create and publish a Salesforce connection in the organization in which you have installed the managed package.

If your guide applies to a Salesforce object and you want to run the guide from Salesforce, ensure that the Salesforce connection associated with the guide has the unique name as **Salesforce**. To view the unique name, right-click the connection and select **Properties Detail**.

The following image shows the unique name for the connection:



Informatica recommends that you have a single Salesforce connection with the unique name **Salesforce** in the organization.

Note: Do not use the connection with the unique name **Salesforce** within a process that contains other Salesforce connections. Create a new connection with a different name, if needed.

For information about creating a Salesforce connection, see the *Salesforce Connector Guide* in the *Connectors* section of the online help.

Step 3: Create and Publish a Guide

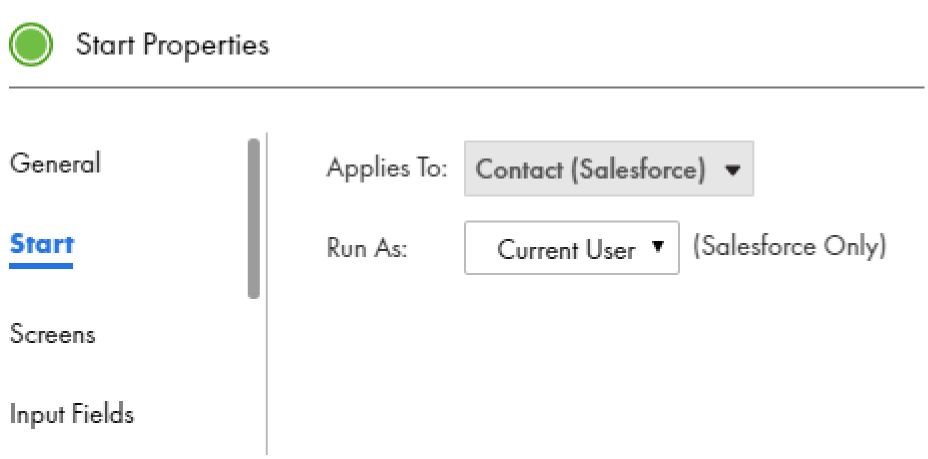
Log in to Application Integration and create a new guide.

If you want to run a guide from Salesforce, the guide must be associated with a Salesforce object.

1. In the properties section of the **Start** step, you must set the **Start** > **Applies to** property to a Salesforce object.
2. Save and publish the guide.

Note: If you set the property to **Home**, the guide appears on your Salesforce **Home** page.

The following image shows a guide that applies to the Salesforce Contact object:

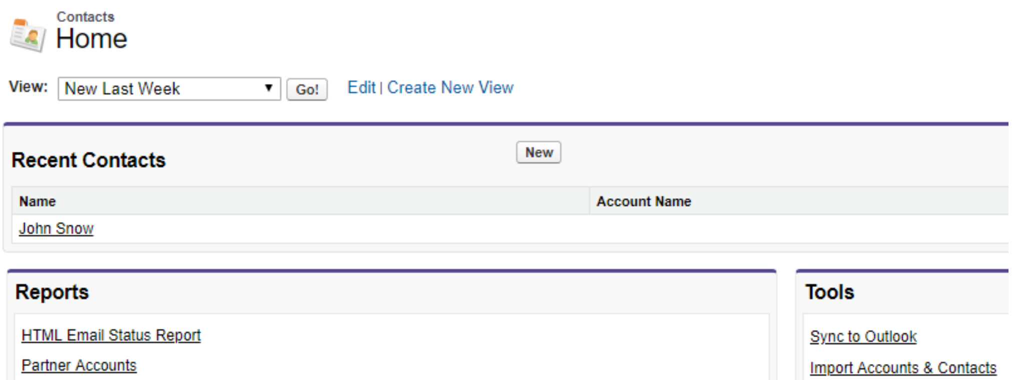


Step 4: Set Up Salesforce to View Guides

To set up the Salesforce page of an object to view related guides, create a section in the home page of the Salesforce object and then add guides to the section.

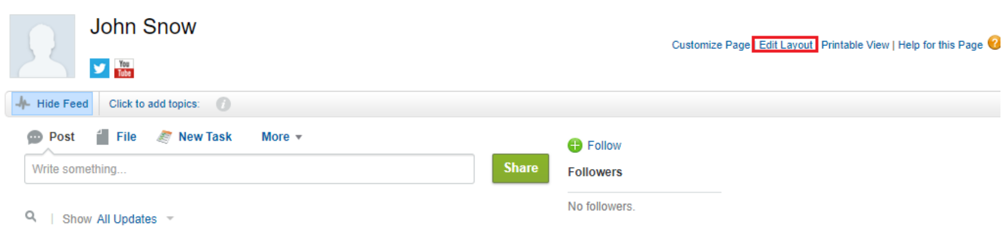
1. Log in to your Salesforce account.
2. Click **All Tabs** and then select the object that you specified when creating a guide.

   The object's **Home** page appears.



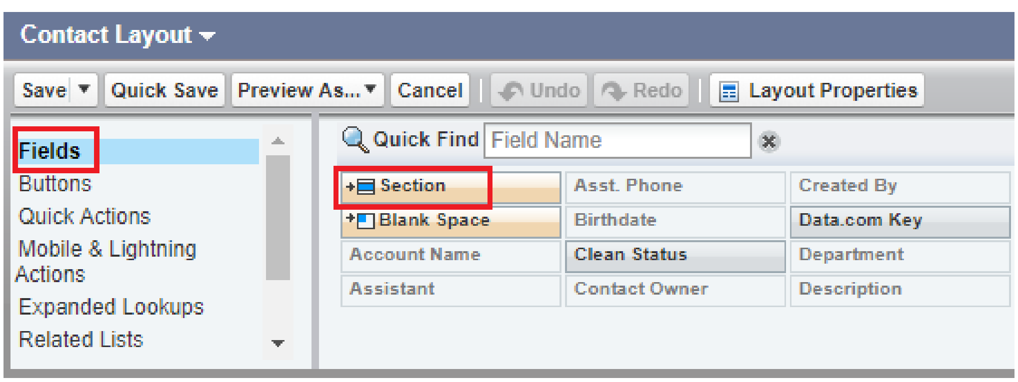
3. Select the item for which you want to see Application Integration guides.

   The item details page appears.



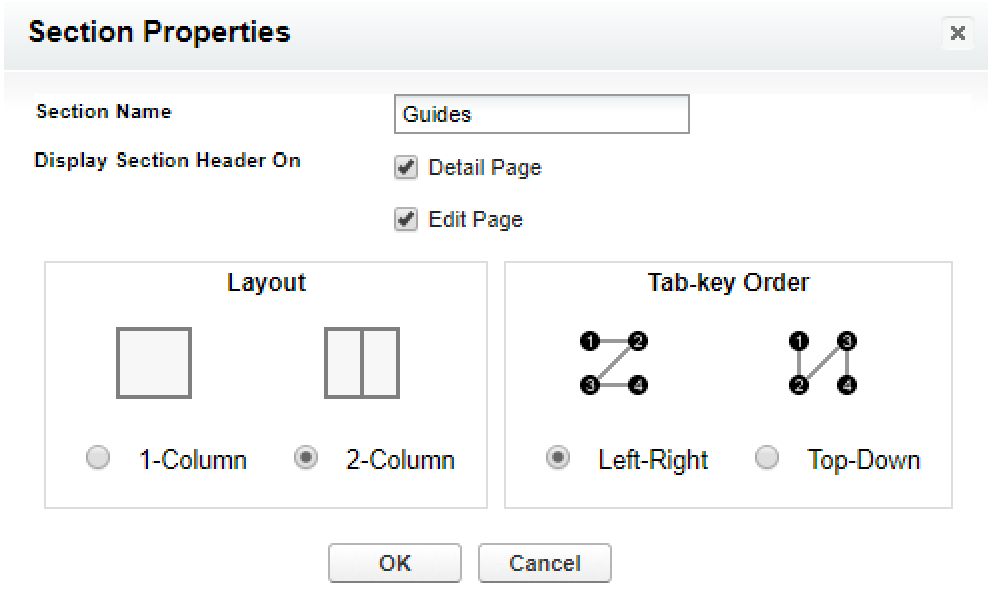
4. Click **Edit Layout**.

   The **Layout** page appears.



5. Drag a **Fields** > **Section** onto the page layout area below.

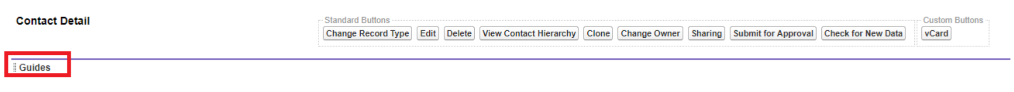
   The **Section Properties** dialog box appears.



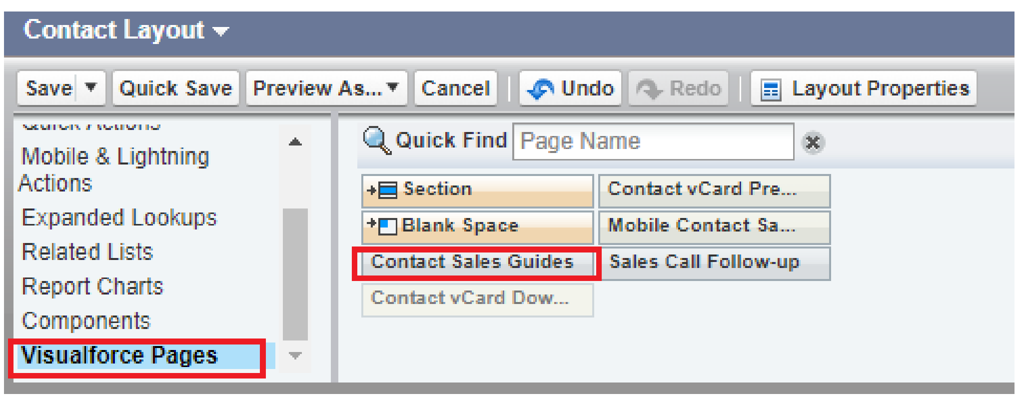
.

6. In the **Section Name** field, enter **Guides** and then click **OK**.

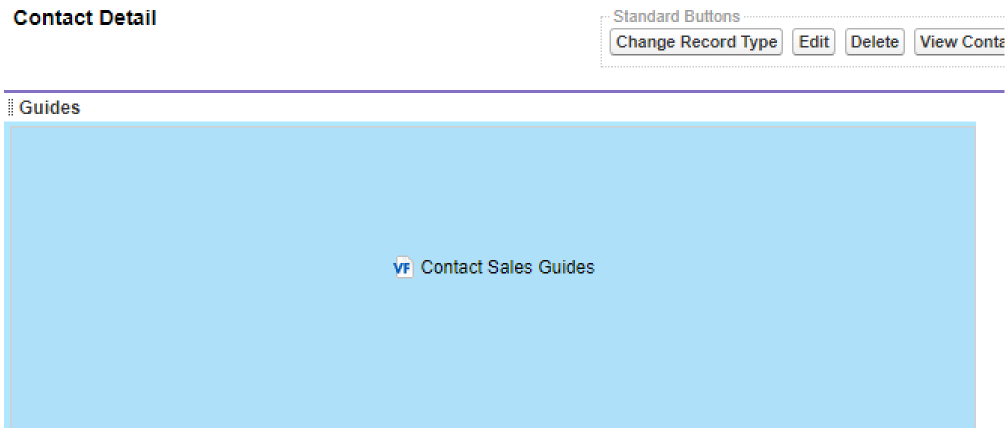
   A section called **Guides** appears in the page layout of the object.



7. From the top of the page, select **Visualforce Pages** and then select a guide.



8. Drag the guide on to the **Guides** section of the page.



All published guides that apply to this Salesforce object appear as links here. Click a link to run the guide.

Connected Apps Guide Integration with Salesforce1 Mobile Applications

With Guide Designer, you can integrate guides with your Salesforce1 mobile applications. This type of integration requires you to use the Salesforce managed package as a Connected App.

Before you start:

- Download the Salesforce1 app.
- Log in to the Salesforce1 app and verify that your environment is working correctly.

The integration requires the following steps:

1. Create a Visualforce page and enable it for Mobile Apps.
2. Create a new Action for the object in Salesforce.
3. Enable Publisher actions in Salesforce.
4. Put the new Action on the Salesforce object page.
5. Verify the Mobile App runs as expected.

Step 1: Create a Visualforce Page and Enable it for Mobile Apps

1. Log in to your Salesforce organization as a system administrator.
2. From the Setup menu, select**App Setup** > **Develop** > **Pages**.
3. Click **New**.
4. Enter a label, for example, aeOpportunityGuideMobileEnabled.
5. Check **Available for Salesforce mobile apps**.
6. Add the Visualforce markup as shown here:

Markup for a Home guide:

```
<apex:page sidebar="false" showHeader="false" standardStylesheets="false"
        cache="true" expires="300" docType="html-5.0" >
 <head>
   <meta name="viewport" content="width=device-width, initial-scale=1,
         maximum-scale=1.0, user-scalable=no" />
 </head>
 <body style="margin: 0px; padding:0px;">
   <icrt:AeSalesGuides sf1="navigation" objectType="_any"
        objectId="{!$User.Id}" extraInfo="Home Guides" nothingOnDone="true"
        match="Role={!JSENCODE(SUBSTITUTE($UserRole.Name, ",", ""))},
             Profile={!JSENCODE(SUBSTITUTE($Profile.Name, ",", ""))}" />
 </body>
</apex:page>
```

Markup for an Account guide:

```
<apex:page standardController="Account" sidebar="false" showHeader="false"
   standardStylesheets="false" cache="true" expires="300" docType="html-5.0" >
 <head>
   <meta name="viewport" content="width=device-width, initial-scale=1,
         maximum-scale=1.0, user-scalable=no" />
 </head>
 <body style="margin: 0px; padding:0px;">
    <icrt:AeSalesGuides sf1="publisher" objectType="Account"
         objectId="{!Account.Id}" extraInfo="{!JSENCODE(Account.Name)}"
         match="Role={!JSENCODE(SUBSTITUTE($UserRole.Name, ",", ""))},
              Type={!JSENCODE(SUBSTITUTE(Account.Type, ",", ""))},
              Industry={!JSENCODE(SUBSTITUTE(Account.Industry, ",", ""))}"/>
  </body>
</apex:page>
```

For other objects, adapt the Visualforce page code for the object you want to work with.

7. Click**Quick Save** to verify that there are no errors.

Step 2: Create a New Action for the Object

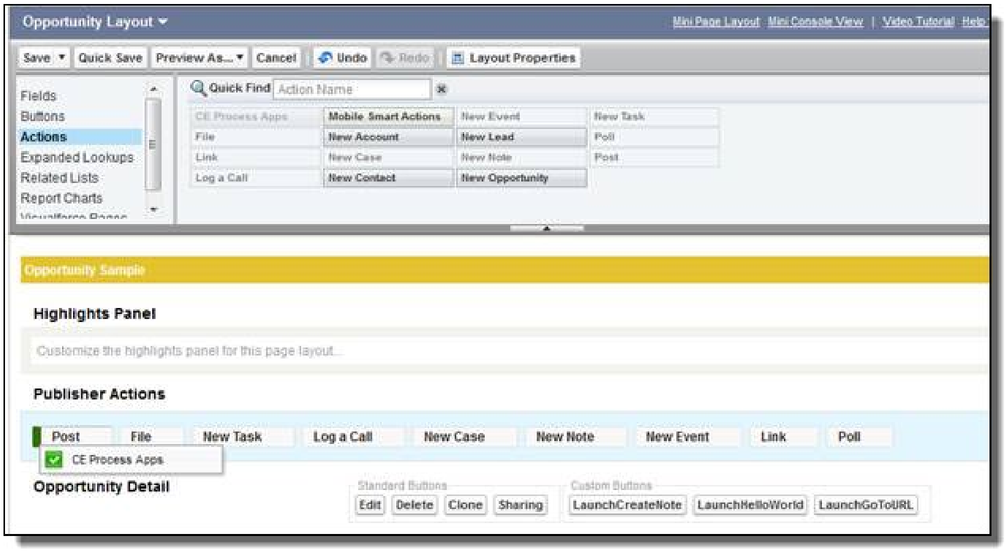
1. From the Setup menu, select the App Setup option for the object you want to work with. For example, **App Setup** > **Customize** > **Opportunities** > **Buttons, Links and Actions**.
2. Click **New Action** and add the Visualforce page created in Step 1.

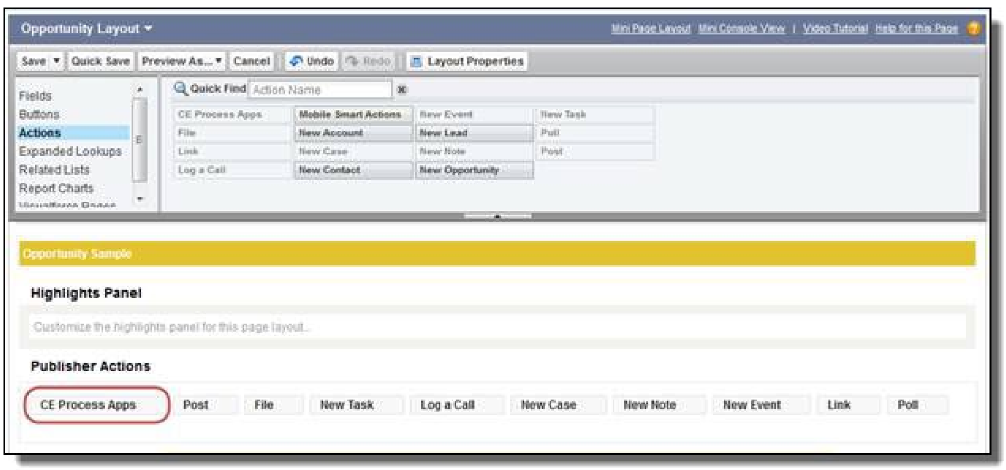
Step 3: Enable Publisher Actions

1. Select**App Setup** > **Customize** > **Chatter** > **Settings**.
2. In the Publisher Actions section, check **Enable Publisher Actions**.

Step 4: Put the New Action on the Object Page

1. Select **App Setup** > **Customize** > **Page Layout** > **Opportunity**.
2. Click **Edit.** Locate the newly-defined action in the Actions list.
3. Drag and drop items to the location within the Publisher Actions section. This is before the Post Action.



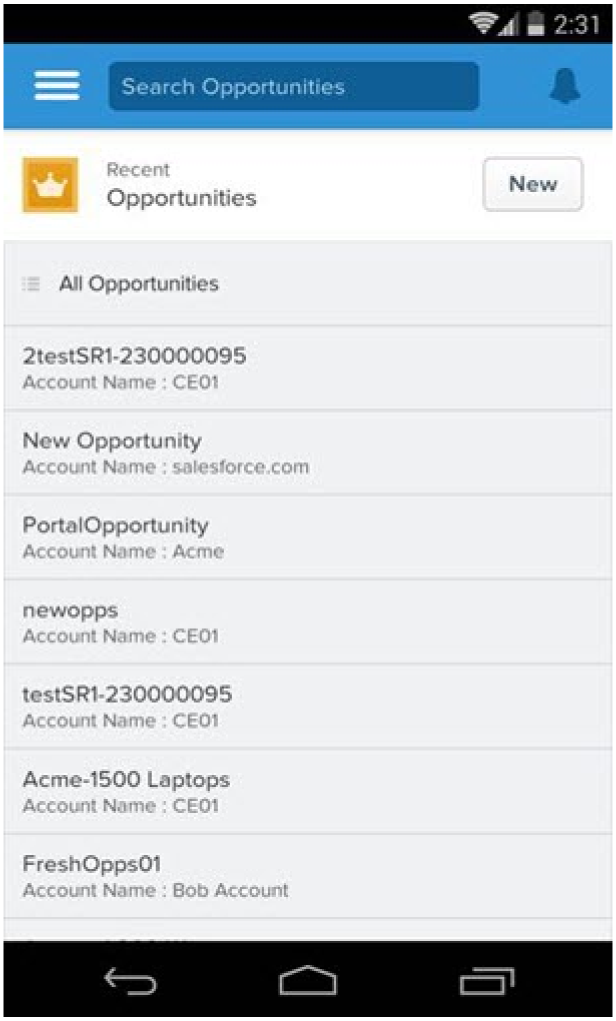


4. Save the layout.

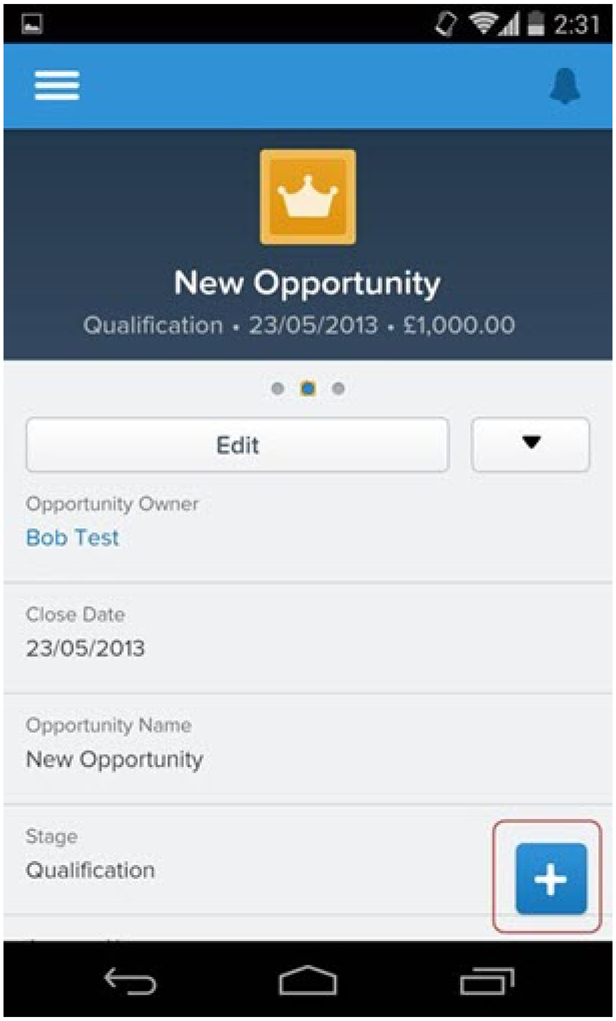
Step 5: Verify

Verify the newly added action in the SFDC1 mobile app.

1. Login to the mobile app.
2. Select an opportunity.

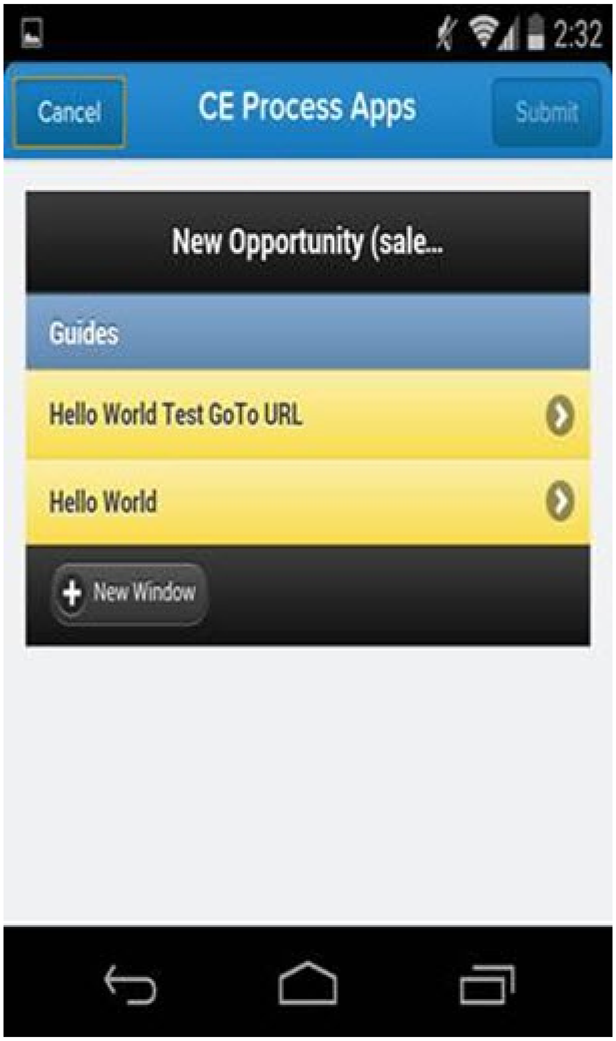


3. Press the "+" button that brings up the publisher actions, which includes the "CE Process Apps" action.

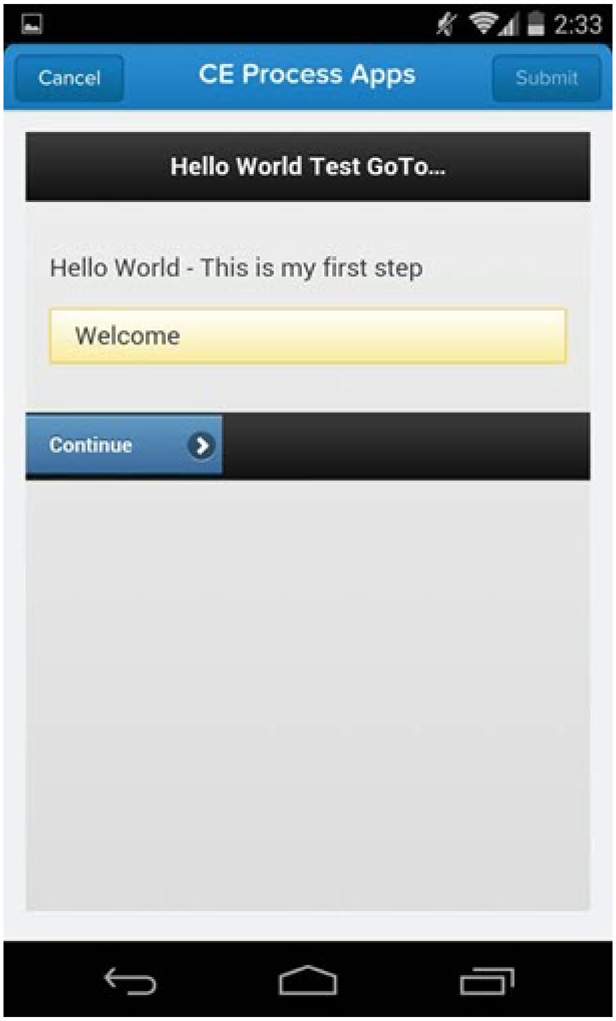


4. Press **CE Process Apps** to display the Designer Visualforce page. It may take a few seconds to load.

You should see the list of guides that are marked to run on a mobile device.


2:32
Cancel
CE Process Apps
Submit
New Opportunity (sale...
Guides
Hello World Test GoTo URL
Hello World
New Window

5. Choose one of those guides to see if it runs successfully.



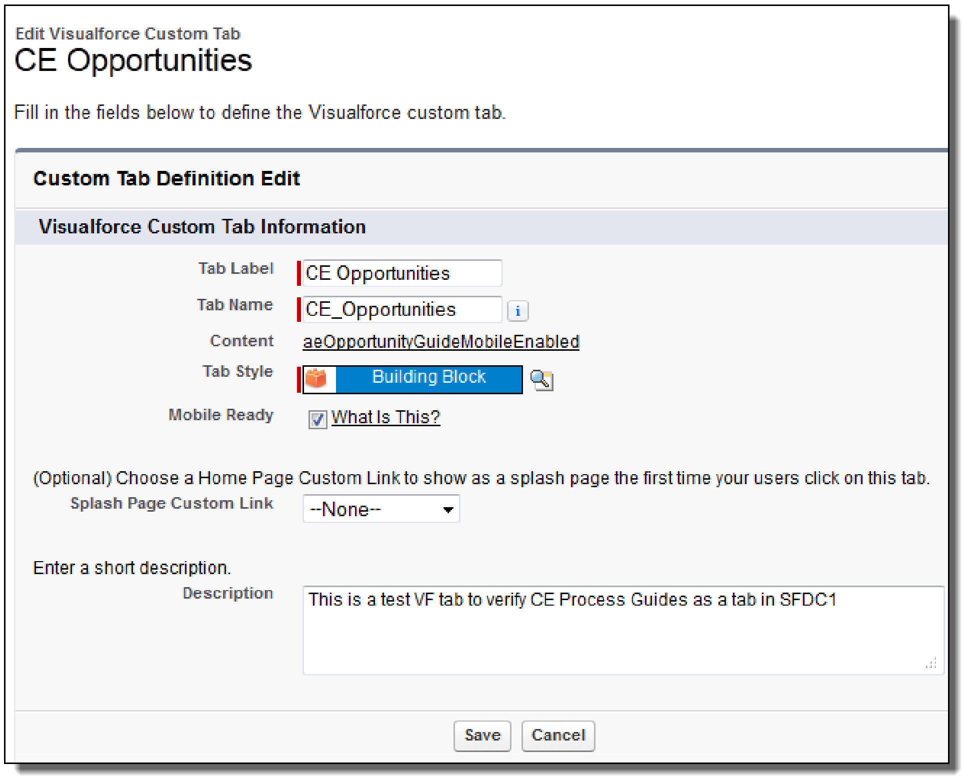
Creating a Salesforce1 App

1. Create a Visualforce Tab
2. Add it to the Mobile Navigation menu.
3. Add a new tab and add the Visualforce page you just created.
   a. Select **Setup > Create** > **Tabs**.
   b. In the Visualforce Tabs section, click **New**.
   c. In the Visualforce Page drop-down list, select the page you created in the previous step.
   d. In the Tab Label field, enter a label name. Note that, this label appears on the navigation menu in Salesforce1.
   e. Click into the Tab Style field, and select some a style. The icon for this style appears as the icon for the page in the Salesforce1 navigation menu.
   f. Click **Next**.
   g. Click **Next** to accept the visibility defaults.

h. Deselect the **Include Tab** checkbox so that the tab isn't included in any of the apps in the organization. You only want this tab to appear when users are viewing it on their mobile device.

i. Click **Save**.

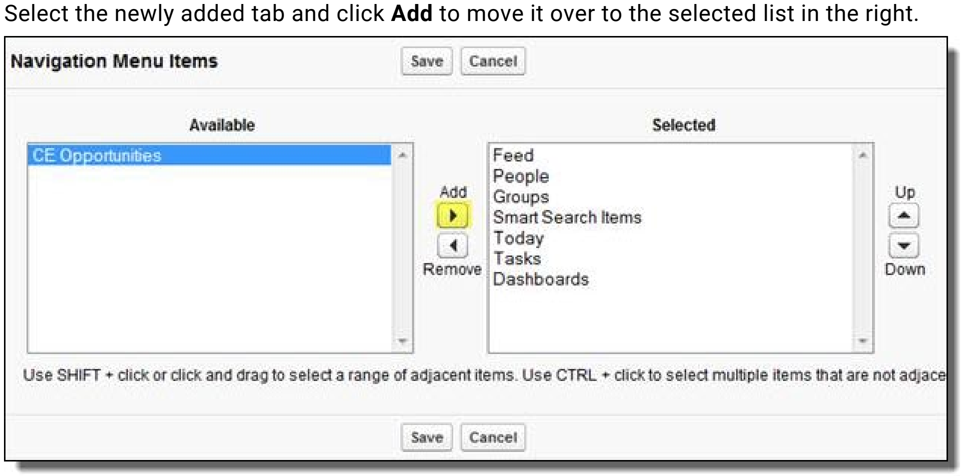
j. On the Custom Tabs screen in the Visualforce Tabs section, click **Edit** next to the new tab that you created..



4. Add the new tab to the navigation menu. Here, it is added as a navigation menu item. It will instantly become available to Salesforce1 app users that have access to it.

   a. Select **Setup > Mobile Administration** > **Mobile Navigation**.

   b. Select the newly added tab and click **Add** to move it over to the selected list in the right.



   c. Click **Save**.

5. When your Salesforce1 users log in, they'll see the CE Opportunities menu item in the navigation menu.

6. Test it out by opening the app. You should be able to open your side menu in Salesforce1 and see your Visualforce Tab in this menu
   Note: You might need to clear the cache for the app if the page does not display immediately.

Accessing Data in Other Salesforce Organizations

After the Salesforce connector is added to your managed package, you can access Salesforce objects from within guides and processes. If you need a guide or process in one Salesforce organization to access data in another, perform these additional configuration actions:

- Perform guide setup in the organization where you want to run the guides.
- Create an additional connection for any other Salesforce organization you need to access.
- In the guides that use Salesforce objects, the objects are from the primary guide's organization. Use the other connection(s) you added to access objects from the other organizations in your guides or processes.

CHAPTER 3

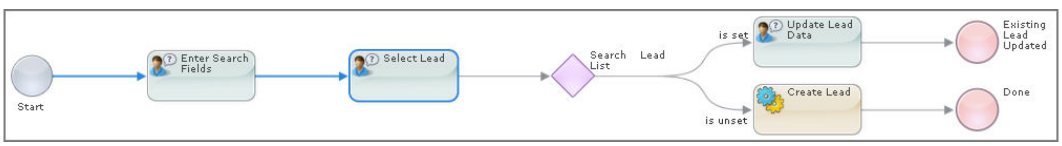
Techniques

This chapter includes the following topics:

Checking for Duplicate Leads and Contacts

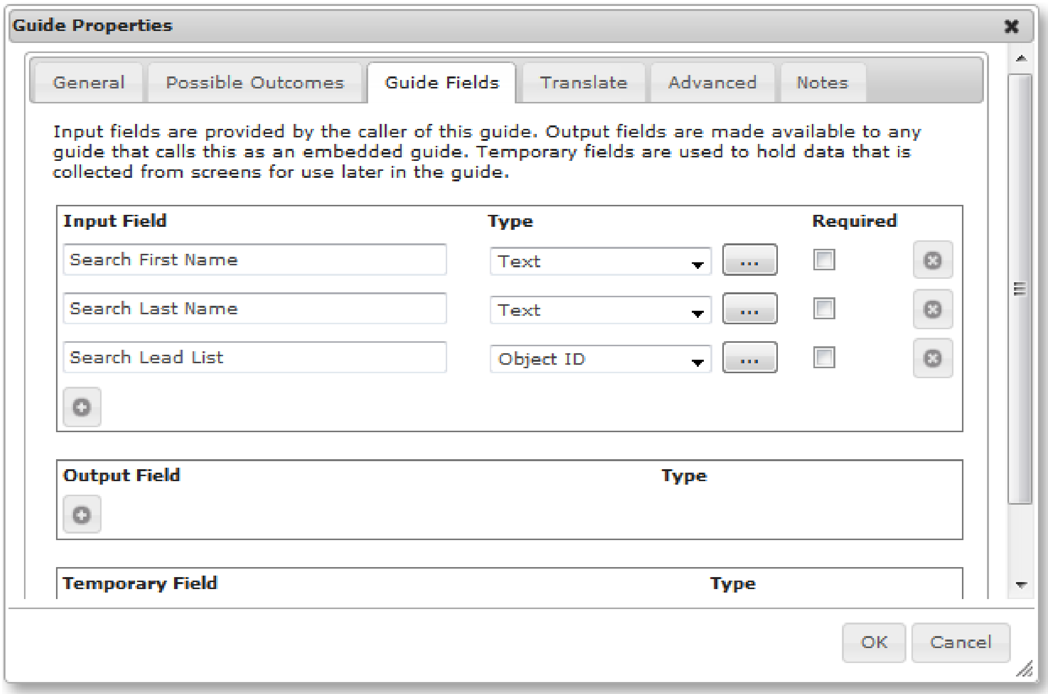
When entering a new lead or contact, you may want to check if it already exists. This topic demonstrates how to do this.

Here's what the completed guide looks like:



Step 1: Set Guide Properties

This is really the crucial step. The guide must access data of some kind but the fields do not exist into which they can be entered as the guide will not be associated with an object. You can create the fields that Guide Designer will use to store this data by using the *Guide Fields* tab.



Now that these fields exist, you can insert them in steps and the data that users enter can be used in searches.

The three fields created are:

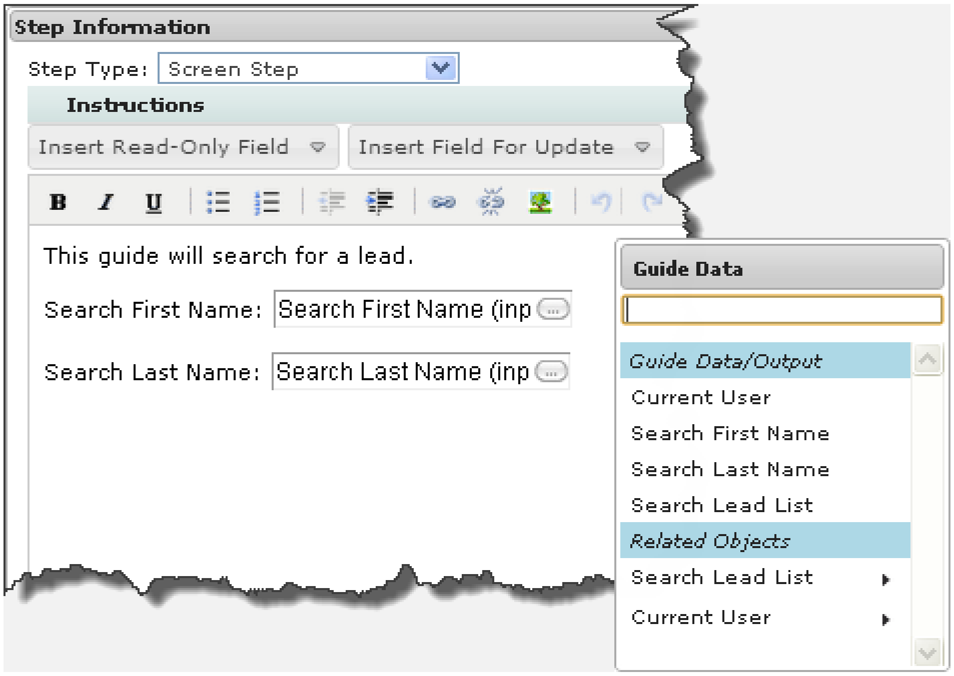
- Search First Name
- Search Last Name
- Search Lead List

Notice that the type of the first two is Text and the last is Object ID. Guide Designer searches for the name. If it is found, it also has the object ID associated with the name. This means that the person was found.

You will want also want to set to create a guide that is not directly associated with a Salesforce object. Also, you should place it on the Home page so that users will have a place from which they can launch it.

Step 2: Create a Screen Step for entering a name

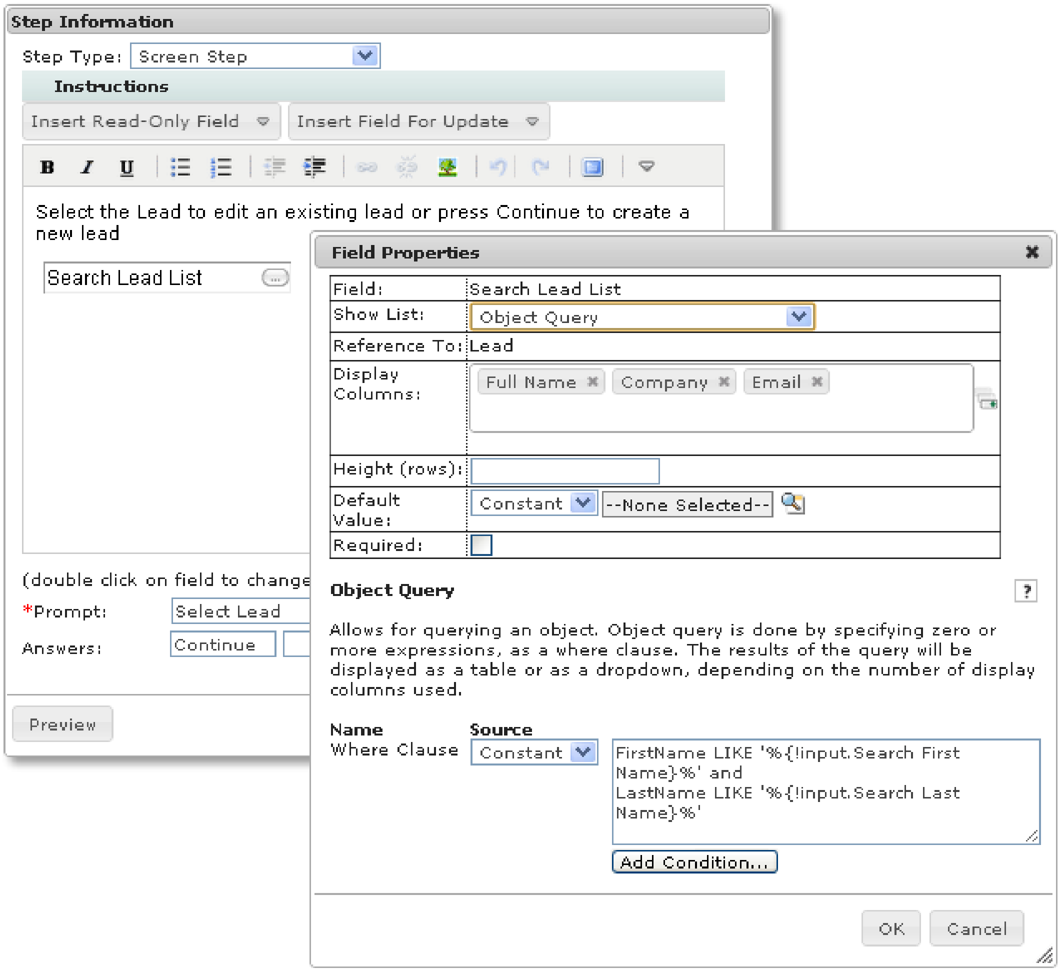
Create a Screen Step into which a user can type a first and last name.



Insert two updateable fields using the picklist. Notice that the three fields defined in the *Guide Fields* tab are there.

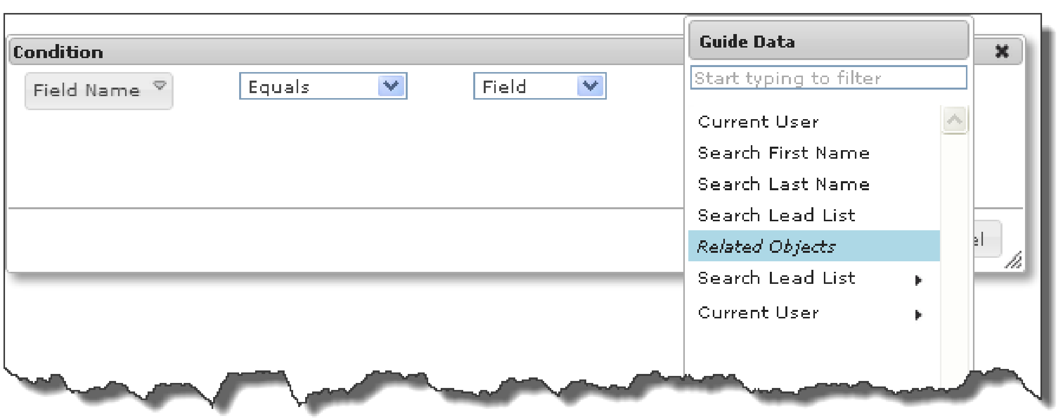
Step 3: Create a screen step for selecting a lead

In this step, you will need to set an inserted field's properties, as is shown in the next figure.



The step will display a list whose contents come from an object query. This query uses the contents of the fields that were filled in within the previous step as part of the query's Where clause.

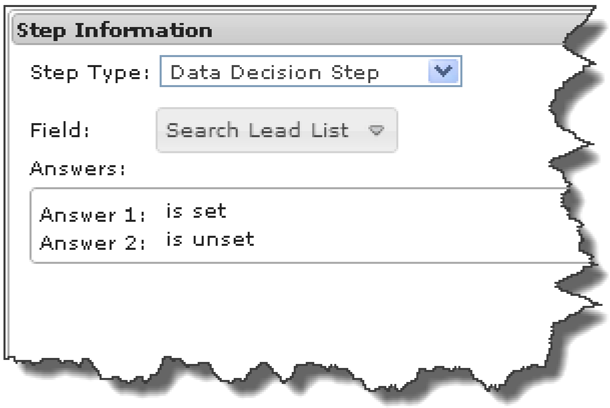
Don't be concerned about the information within the curly braces "{}". If you use the **Add Condition** dialog, Guide Designer adds them for you.



After Guide Designer executes this step, it will show a four-column table. It will have the three columns entered within the Display Columns area and a fourth column containing radio buttons. If the lead is within the table, the user will select one of them.

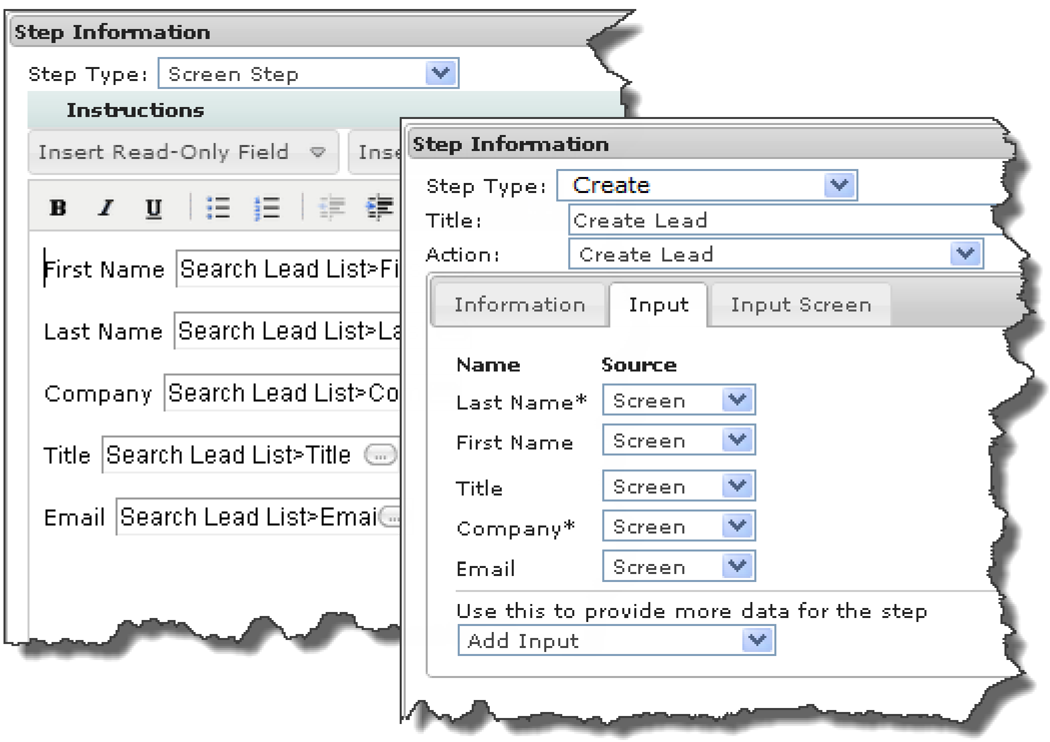
Step 4: Was something returned?

After the previous step executes, it sets the value of the Search Lead List field. You can now add a decision step that tests if this field was "set" or "unset", indicating if something was or was not found.



Step 5: Next Steps

All that is left is create the steps that follow the decision step. In this example, one branch is followed when the lead does not exist and the other for when it does.



Email Actions (Salesforce)

This topic describes the email actions that Salesforce provides.

Send Email Using Template and Send Free-Form Email

The **Send Email Using Template** and **Send Free-Form Email** actions are very similar as they both send email. The difference between them is that a template contains the subject and body of the email while the free form requires you to add them.

- When using a template, you will need to specify either the object to take information from or the email address, but not both. If neither is specified, the email is sent to the address within the current object.
- If there is no template, you will need to add a subject and the email's body. Specify either the object or the email address, but not both. If neither is specified, the email is sent the current person.

Note: Unlike other steps, email steps cannot be rolled back either by selecting a step in the History or by you backing up to it. This is because the email is already sent. Also, if you do step back to this step or one executed prior to it, the email will again be sent.

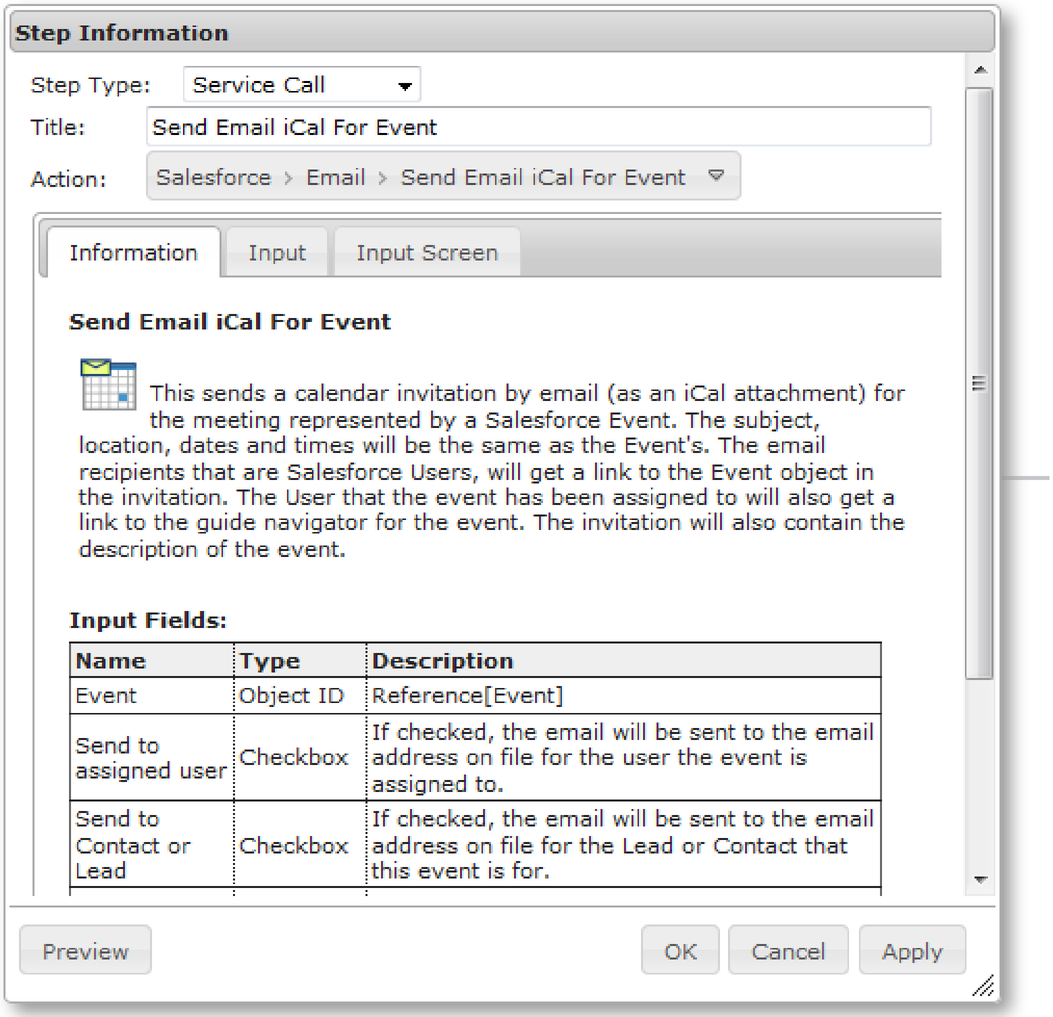
Using **Send Free-Form Email** is often better than **Send Email Using Template** even when there's a template that meets your needs because using a template doesn't give your users any flexibility. However, there's a way to use a template and allow users to modify things. Do this by using the **Send Free-Form Email** Service Call step, and setting the source of email's body to **Screen**. Next, double click on the body's input field and then insert the email template so that it is the default value of the field. This default value can include fields of any object that is available from the guide. Your users will see the template shown in an HTML editor inside the step with its data inserted for field values used by the template.

You can add attachments to an email. While all attachments are files, Salesforce differentiates attachments by where they exist and how they are used:

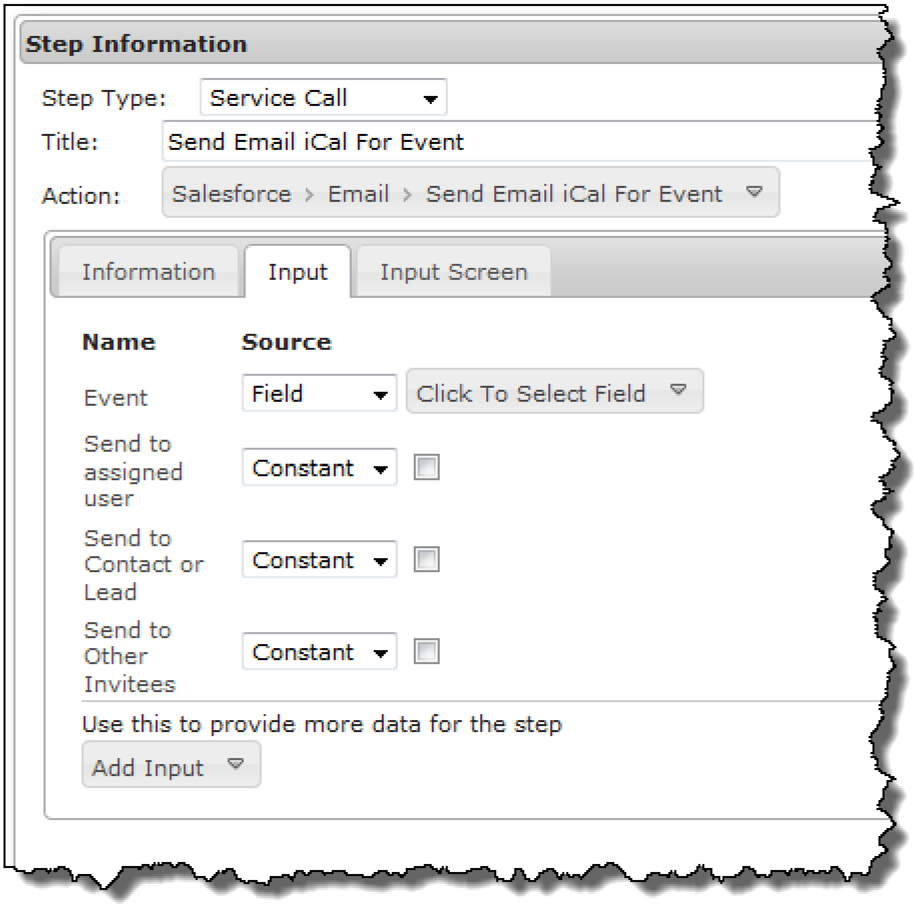
- **Attachment Attachment**: The files named in Salesforce attachment lists. These lists appear in the detail pages of some objects. These attachments can include feed attachments.
- **Content Attachments**: Shared and published corporate files. These files are typically shared with co-workers or sent to customers. For example, they could be Google Docs.
- **Document Attachments**: Web resources that others from your site have placed on the Web. Some of the kinds of information that are contained within documents attachments are logos, graphics, and template files.
- **File Attachments**: Files that you upload. While these are initially private files, they can be shared with co-workers.

Email iCal for Event Action

This action sends a calendar invitation by email (in iCalendar format) for the meeting created as a Salesforce Event. The dates and times are the same as the dates and times of the event. The email recipients who can log in to your Salesforce site receive an iCalendar file that contains a link to the **Event** within Salesforce. The Salesforce user to which the event is assigned receives a link that when clicked starts the guide for the event.

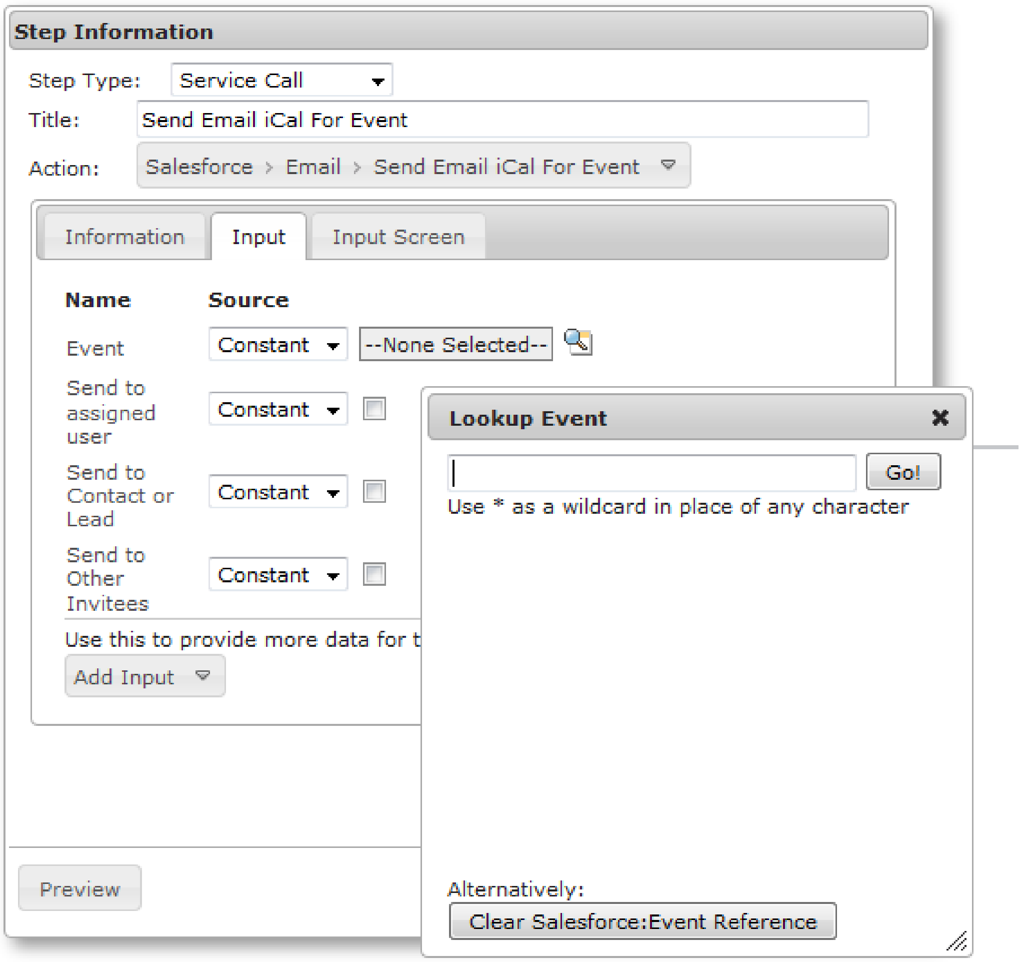


The **Email iCal for Event** Service Call step uses up to six fields. The first four are required. The final two are optional. The following figure shows these fields on a step's Input tab.


Step Information
Step Type:
Service Call
Title:
Send Email iCal For Event
Action:
Salesforce › Email › Send Email iCal For Event
Information
Input
Input Screen
Name
Source
Event
Field
Click To Select Field
Send to assigned user
Constant
Send to Contact or Lead
Constant
Send to Other Invitees
Constant
Use this to provide more data for the step
Add Input

These fields are:

- **Event**: The object ID that is a reference to an event. Process Designer will provide this ID. In this figure, the ID is that of the current Activity ID. You can, of course, select another field; for example, you can also select the ID for other events.



  You can also use a query to select the event.

- **Send to assigned user**: If checked, the email is sent to the email address stored within Salesforce for the user to which the event is assigned.
- **Send to Contact or Lead**: If checked, the email is sent to the email address stored within Salesforce for the lead or contact that this event is for.
- **Send to Other Invitees:** If checked, the email is sent to the email addresses stored within Salesforce for the users, leads or contacts that are listed for this event.
- **Other Recipients**: A semicolon separated list of email addresses of other people you would like to attend the event.
- **Body**: Optional text that is inserted into the email after any links that are sent.

CHAPTER 4

Customizing Salesforce

This chapter includes the following topics:

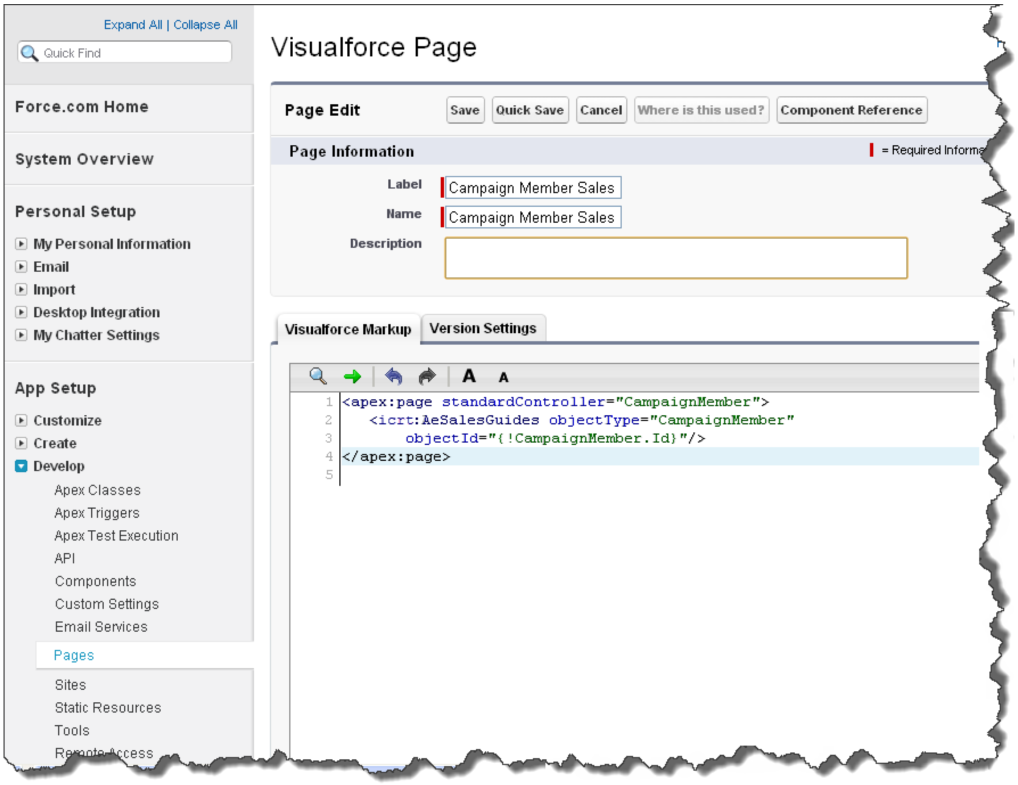
Customizing and Creating Salesforce Pages

You can use Visualforce page definitions for most of the common Salesforce objects. However, you may use other objects from which you want to launch guides or you may create your own. This topic discusses how to create and use a Visualforce page. This example will use the CampaignMember object. This process is straight forward and only contains three steps. If you are using version 27 of the Salesforce API, please read the note at the end of this topic.

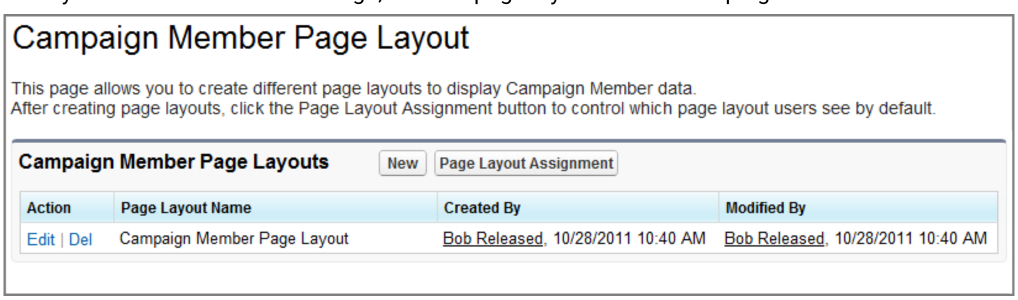
Note: See "Adding Guide Areas to Salesforce Pages" on page 50 for a more complete discussion as this section omits some of the details shown there.

1. Create the Campaign Member Sales Visualforce Page. The markup you will add is:

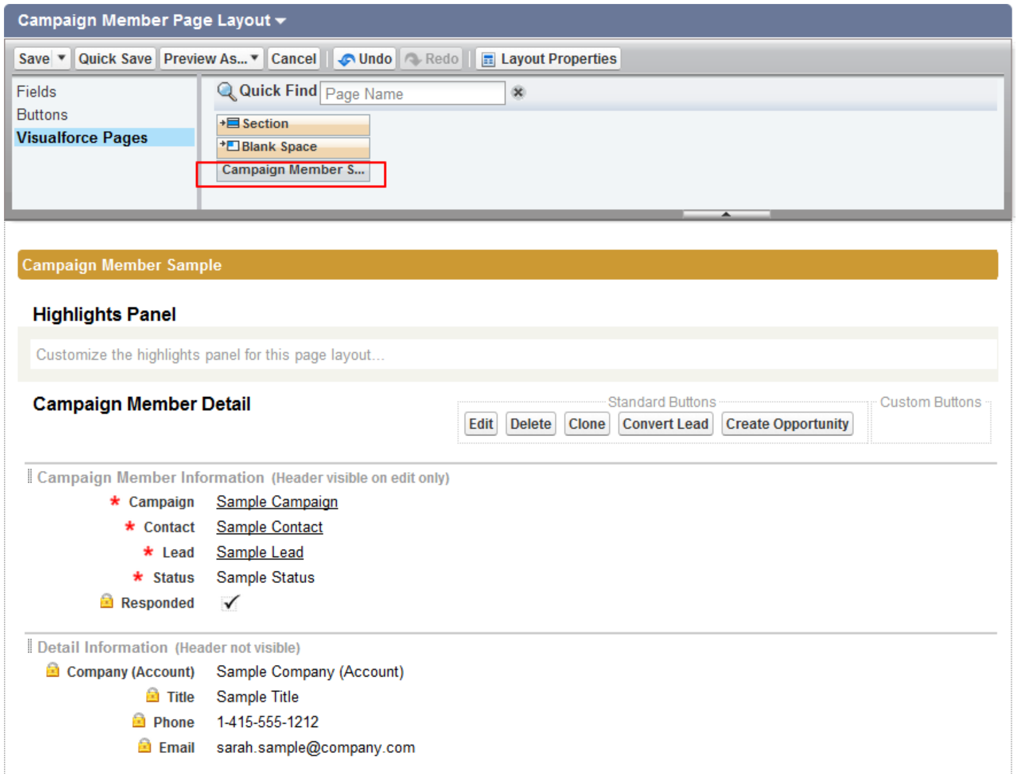
```
<apex:page standardController="CampaignMember">
     <icrt:AeSalesGuides objectType="CampaignMember" objectId="{!
CampaignMember.Id}"/>
</apex:page>
```



2. After you save the Visualforce Page, edit the page layout for the Campaign Member:



3. Add the new page to a section in the page layout:



Salesforce API, Version 27

If you are using version 27 of the Salesforce API and are creating a Visualforce page to embed an `AeSalesGuides` Visualforce component using code similar to the following:

```
<apex:page standardController="Account">
   <c:AeSalesGuides objectType="Account" objectId="..."/>
</apex:page>
```

Salesforce displays the following error:

```
Error: <apex:attribute assignTo> cannot be same as the <apex:attribute name> ( objectId )
```

This error prevents you from creating new Salesforce pages that have guides.

The following is copied from
http://salesforce-walker.blogspot.com/2013/02/sprin-13-release-notes-in-visualforce.html:

Compile-Time Checking for Custom Component Attribute Names:

In the Spring '13 release, there is a new compile-time check that prevents giving custom Visualforce component attributes the same name as the assignTo value (which can refer to a property or method in the component's controller). This checking prevents a number of programming errors (bugs) that can result in runtime errors. This change is versioned, and you must update your Visualforce components and the pages that contain them to API version 27.0 to enable the new behavior. Previously, it was possible to create a custom component with an attribute that had the same name as the `assignTo` value.

For example:

```
<apex:component>
<apex:attribute name="currentpagename" type="string"
    assignTo="{!currentpagename}"
    description="This component can't be saved."/>
```

This component is no longer allowed.

```
</apex:component>
```

Force.com Visualforce:

To make this component compile under API version 27.0, change either the attribute name or the assignTo value (which might require you to also change a controller method or property). For example:

```
<apex:component>
<apex:attribute name="thepagename" type="string"
         assignTo="{!currentpagename}"
         description="This component saves just fine."/>
This component is OK!
</apex:component>
```

Note: Both components, as well as the pages they are displayed on, must be updated to API version 27.0 to enable the new compile-time check.

Attributes to Customize Salesforce Pages with Guide Designer

You can use the following attributes when interacting with Visualforce pages in Guide Designer. For example, see "Customizing and Creating Salesforce Pages" on page 39

Name	Type	Description
alwaysNewGuide	Boolean	Using this flag avoids any correlation with other running guides so guides that open always start from the beginning.
closeOnDone	Boolean	True if a the window is closed when a user clicks done. This is used when a guide is launched in its own window.
entryIds	String	Comma-separated list of entry IDs to be displayed.
extraInfo	String	For reporting purposes, extra info to be displayed in Informatica Cloud Real Time reports goes here.
guideName	String	Names the guide that will be displayed.
height	String	Optional height of the guide area iframe. If you do not use this attribute, the default will be 100 percent of its container.
id	String	An identifier that allows the component to be referenced by other components in the page.
match	String	Comma-separated list of tags that are displayed. Guides that don't contain one of these tags are filtered out of the list for the user. This allows for both static and dynamic filtering as tags can be field values for example `Status:Warm`.
nothingOnDone	Boolean	Indicates that the browser page is to stay where it is when the user hits Done. Note , however, the tab for the finished guide is closed.
objectId	String	The ID of the object to be used as the context of the guides; for example `Account`.Id for use accounts. This parameter is required.

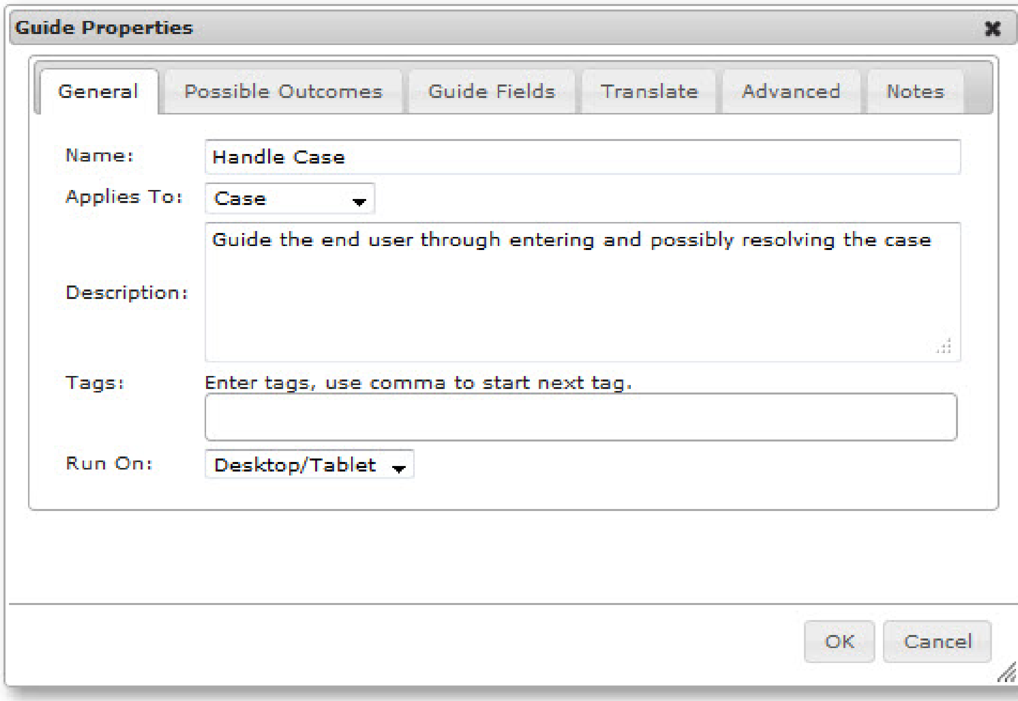
Name	Type	Description
objectType	String	The type of record the guides is used with; for example, set to `Account` when used with accounts. This parameter is required.
openIfOne	Boolean	If there is only one guide to display, open it and do not show the Guide Navigator. **Note:** This is always true if a `guideName` attribute is set, regardless of this setting.
rendered	Boolean	A Boolean value that specifies whether the component is rendered on the page. If it is not specified, this value defaults to true.
showCategories	Boolean	Turns displaying tags as categories on when true. Tags containing equal signs and tags beginning with a period are never shown.
topLevelWindow	Boolean	If false, shows the 'open in new window' link. True means the Guide Navigator is a top level window. Default is false.
useResponsive	Boolean	Using this flag when loading a named mobile guide on a desktop or tablet allows it to run.
width	String	Width of the guide area iframe. If you do not specify a value, the default is 100 percent of its container.

Creating a Button or Link to Launch a Guide

This topic describes how to create a button or a link that launches a guide from a Salesforce object. The steps for doing this are:

1. Create a Visualforce page.
2. Update the Visualforce page's security settings.
3. Create a button (or link) for the desired Salesforce object.
4. Add the button (or link) the object's page layout.

An example is used to show these steps. In this example, a button is added that launches the Handle Case guide. Here is the General tab of the **Guide Properties** dialog for this guide:

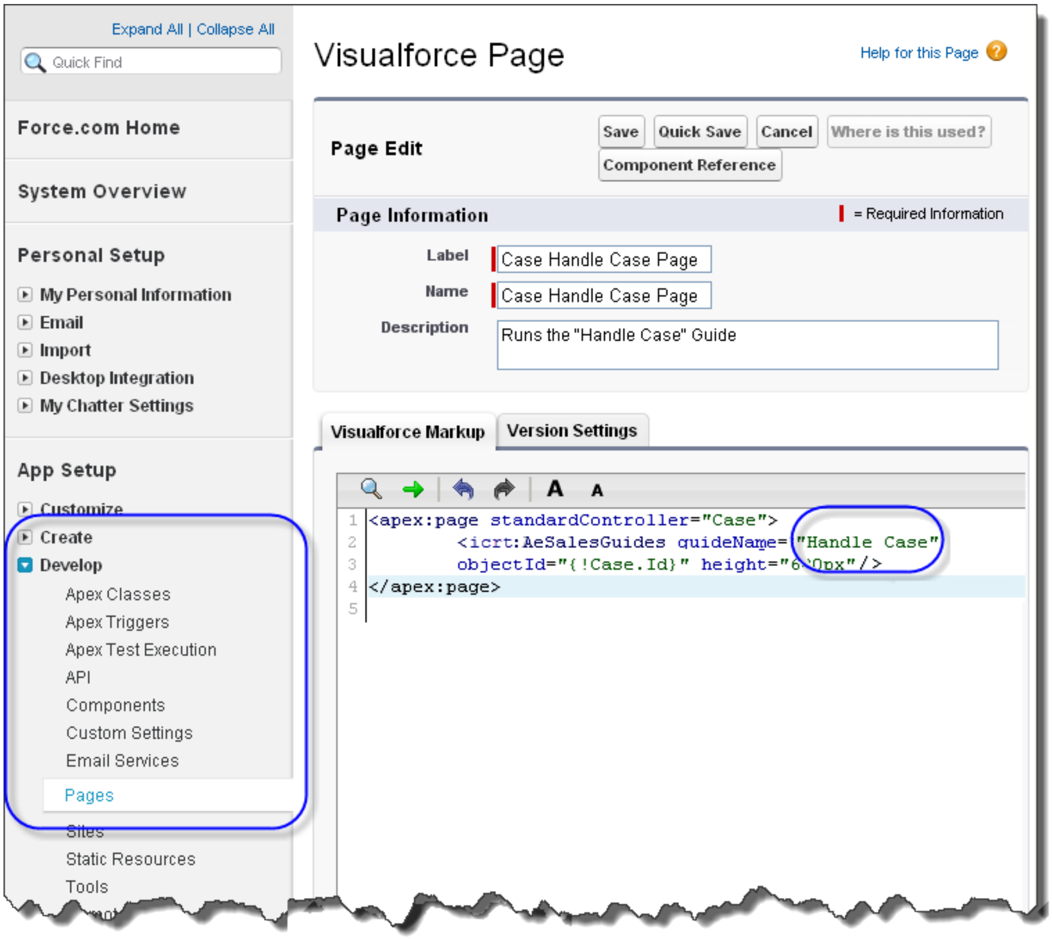


Creating a Visualforce Page

Set the Visualforce Markup to:

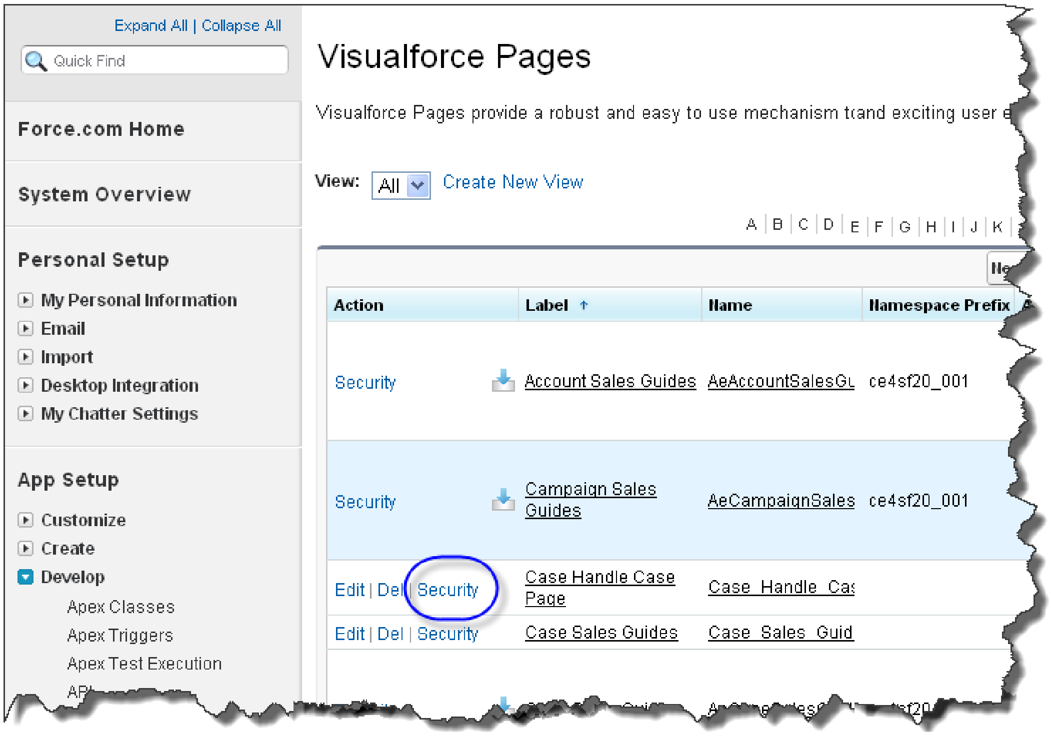
```
<apex:page standardController="Case">
        <icrt:AeSalesGuides guideName="Handle Case" objectType="Case"
        objectId="{!Case.Id}" height="600px"/>
</apex:page>
```

Note: Make sure that the guide's Name value here matches the name set in the Guide Properties dialog.

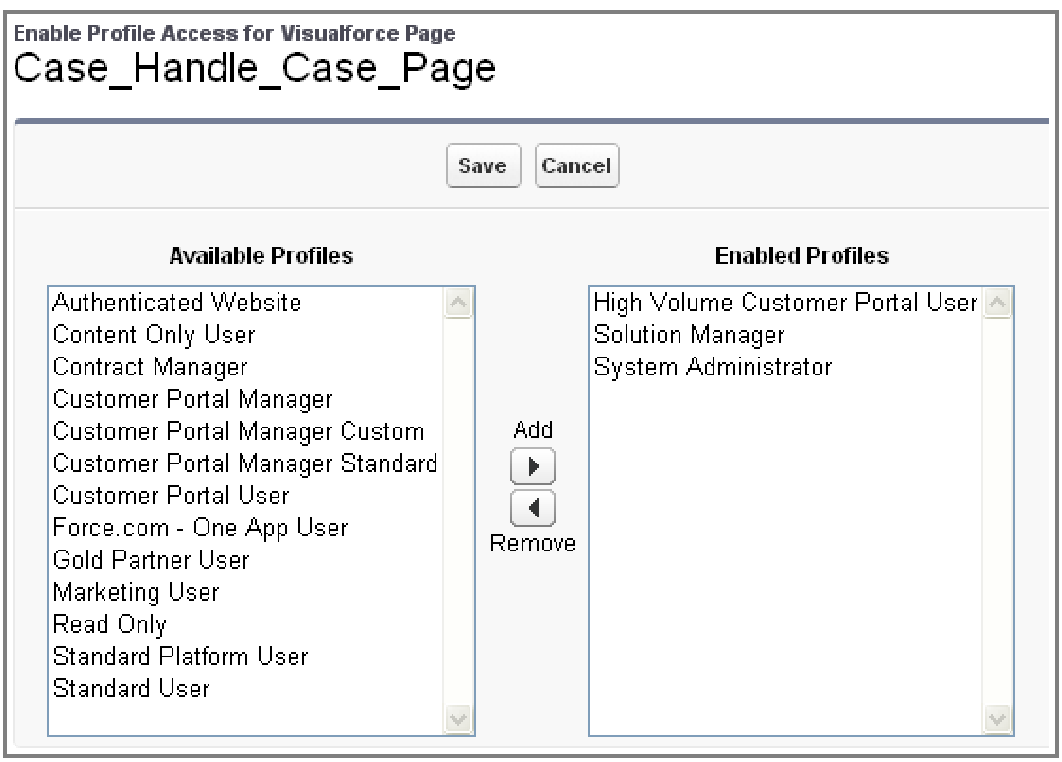


Update the Security for the Visualforce Page

Click the **Security** link for the page you just created:



Add one or more profiles displayed within the Available Profiles list to the Enabled Profiles list.



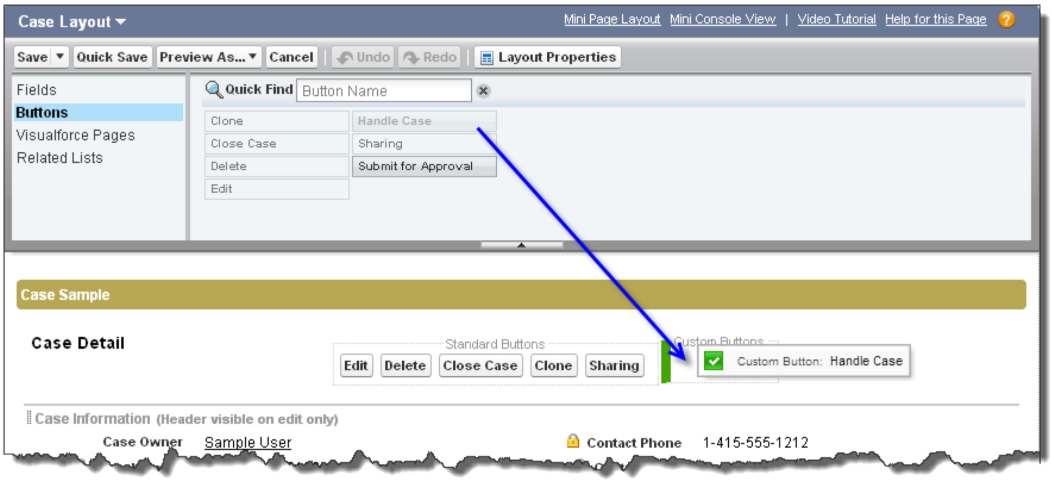
Save your changes.

Update the Page Layout

Select **Cases** | **Page Layouts** on the left. On the Case Page Layout page, click edit to display the page onto which you will be placing the button (or link).

Note: To add custom links to the page layout, select the Custom Links category on the palette and drag one or more custom links from the palette to the Custom Links section on the page layout.

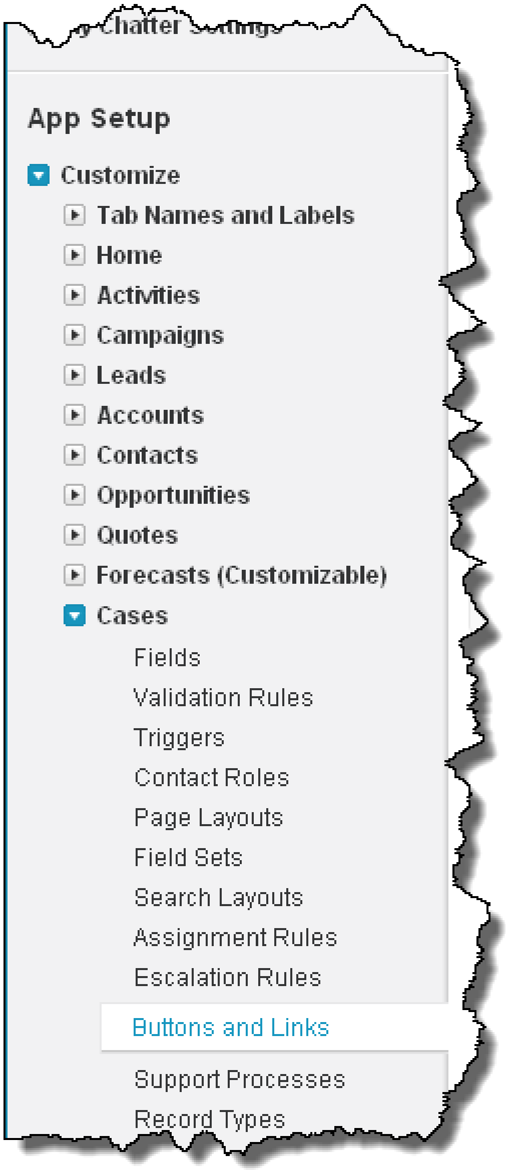
Notice that Buttons is selected on the left, not Visualforce Pages, which is the entry after this one.



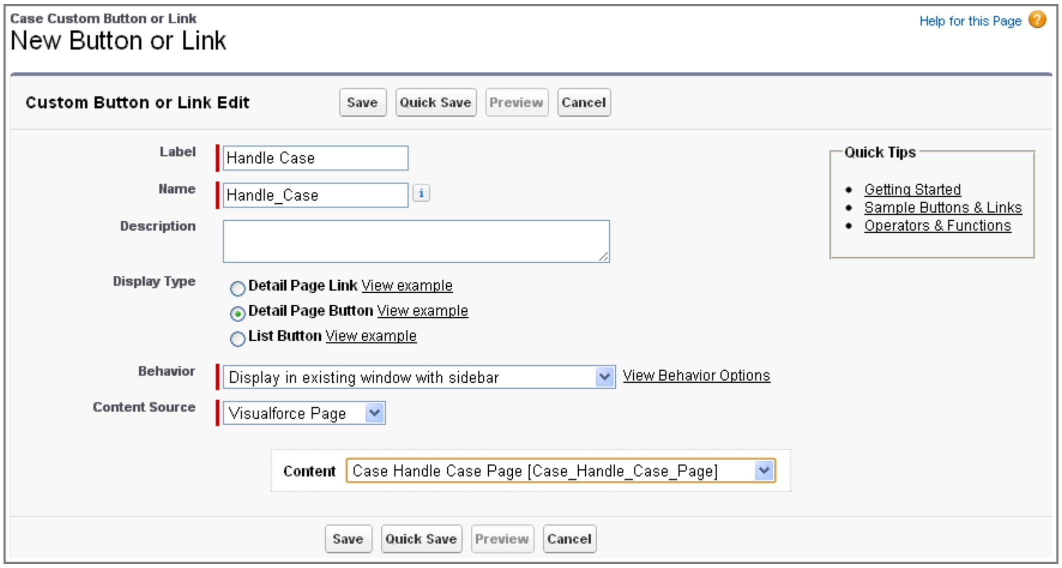
Save your changes.

Create a Button on the Salesforce Page

Within the Custom Buttons and Links area, press New to create a new button.



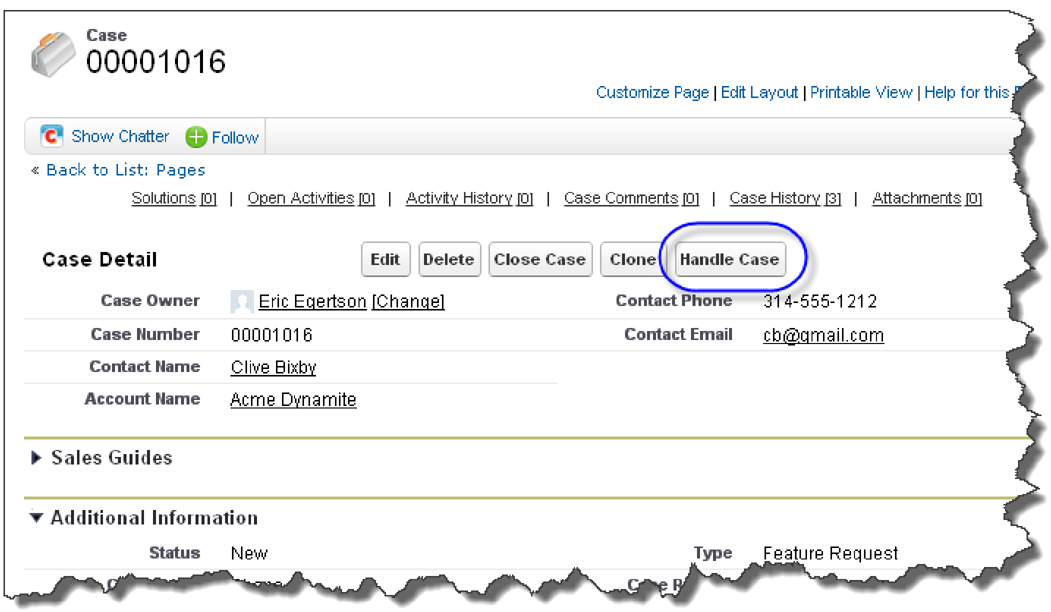
Set the button's details similar to the following. Notice how **Detail Page Button**, **Behavior**, and **Content Source** are set. **Content Source** must be set to the Visualforce page that you created earlier.



Note: The way in which you create a link is similar; just select **Detail Page Link** instead of **Detail Page Button** and then use link in place of the button in the instructions that follow.

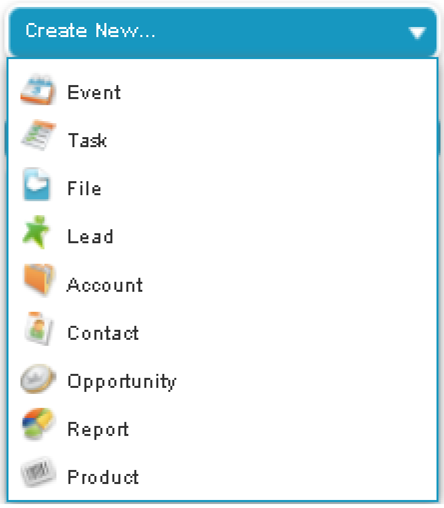
Launch the Guide

Open a case to see the newly added **Handle Case** button.



Running a Guide When the User Clicks New

Salesforce users can create new objects using the **Create New** picklist, a button, or a link.



After users select one of these, Salesforce displays a page into which the user can enter information. If you would prefer that they run a guide, you can override these actions for standard objects. Here's how:

1. Go to that object's *Buttons and Links* setup page; for example, **App Setup Customize > Opportunities > Buttons and Links**.
2. Within the displayed table, find the **New** row, and then click **Edit**.
3. Select the Visualforce page on which you are going to override the New action.
4. Create the Visualforce page that runs the guide. For example, if you wish to run the Create Opportunity guide, you might create a page that runs it such as:

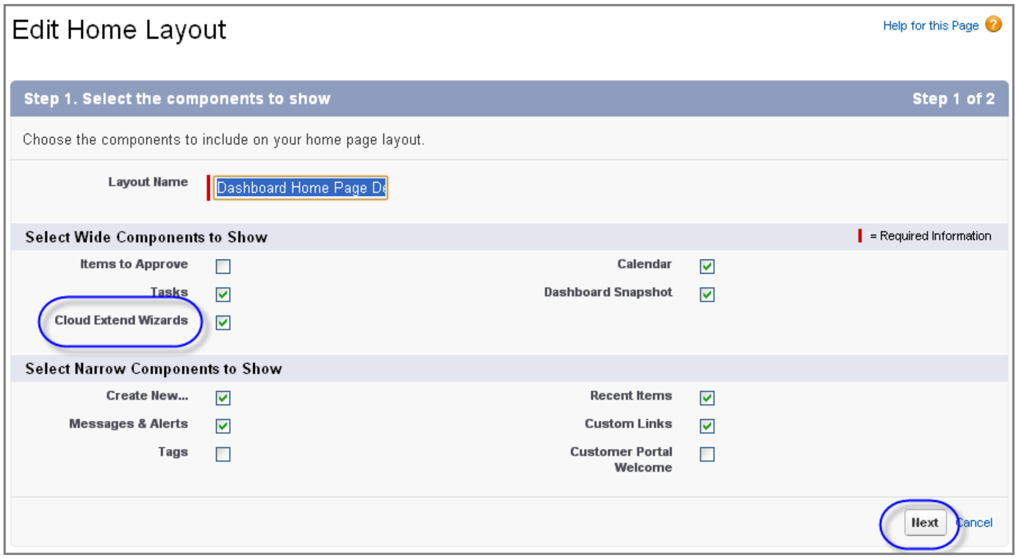
```
<apex:page sidebar="false"  showHeader="false">
    <icrt:AeSalesGuides objectType="_any" objectId="{!$User.Id}"
        guideName="Create Opportunity"/>
</apex:page>
```

Adding Guide Areas to Salesforce Pages

This topic describes how you add Guide Designer Wizards to a user's Salesforce Home page.

Adding a Guide Area to the Home Page

1. Open the Setup menu in the upper right corner of Salesforce. From the displayed page, select **App Setup > Customize > Home > Home Page Layouts**. Both of these are indicated in the following figure.
2. Click **Edit** for to make changes to your home page. You may have more than one page. In the Edit Home Layout page, go to the **Select Wide Components to Show** area, and check the **Guide Designer Wizards** checkbox. Click **Next**.



3. Place the Guide Designer Wizards component in the position where it should display on the page. When you're done, click **Save**.

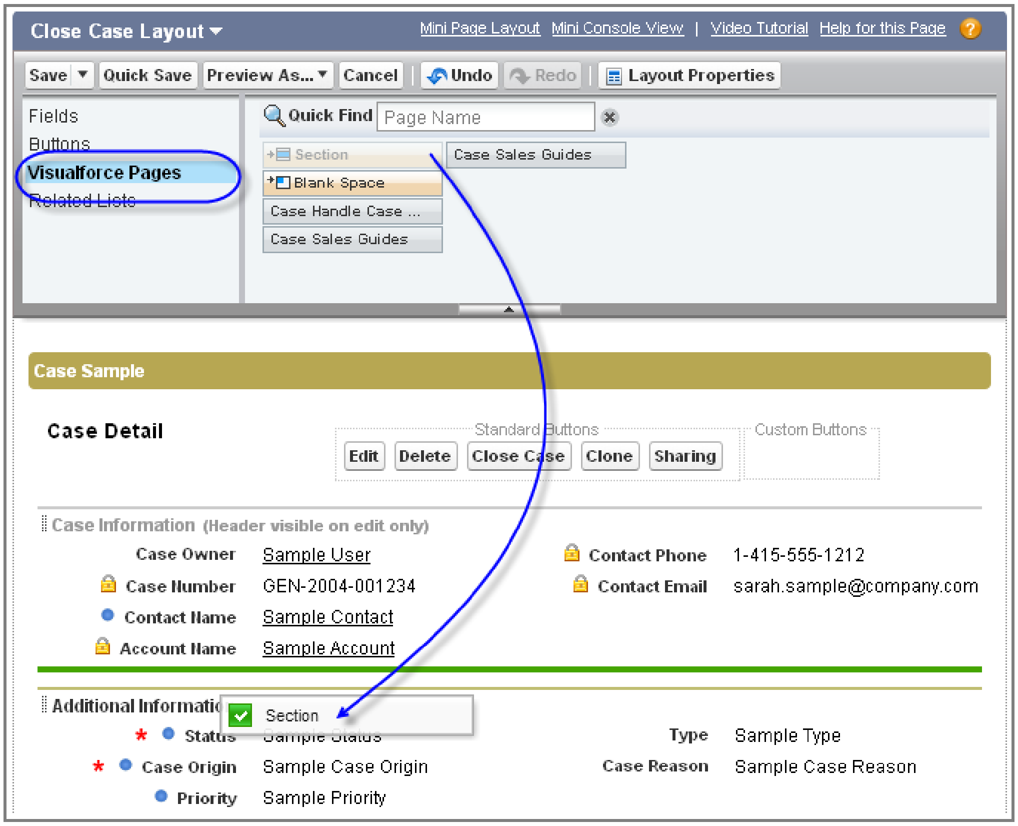
After adding the Guide Designer Wizards component, an area displays for all users who see this page. On a PC or Mac, users do not see wizards whose Run on setting is set to *Smart Phone*. Similarly, smart phone users do not see wizards whose *Run on* setting is *Desktop/Tablet*.

Adding Guides to Salesforce Pages

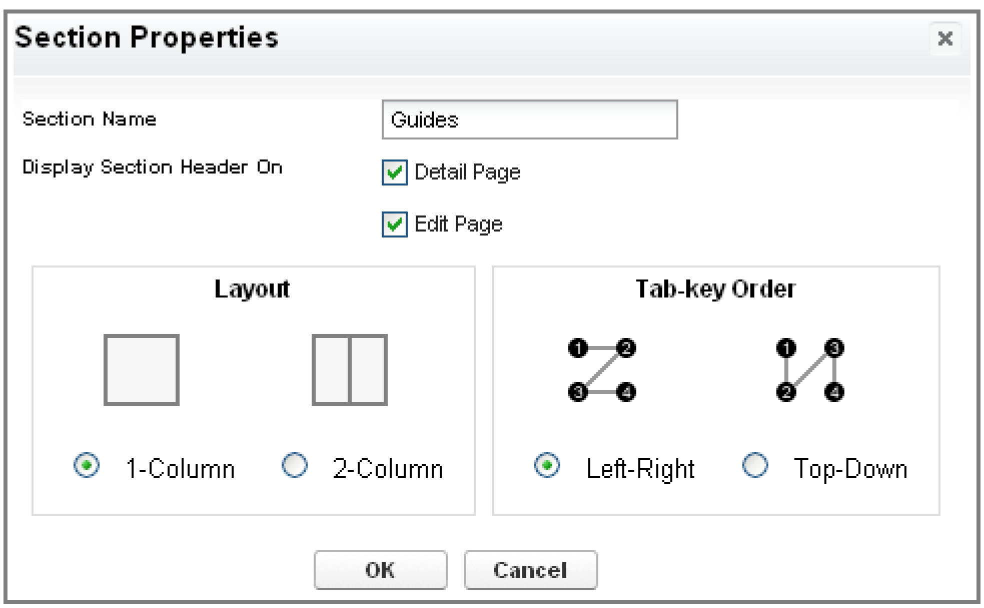
This topic describes how you install a guides area on page; that is, this how you install a guide area on all pages except for the home page.

1. Select **"your Name" > Setup** on any page. From the displayed page, select **App Setup > Customize > Home > Home Page Layouts**. Both of these are indicated in the following figure.

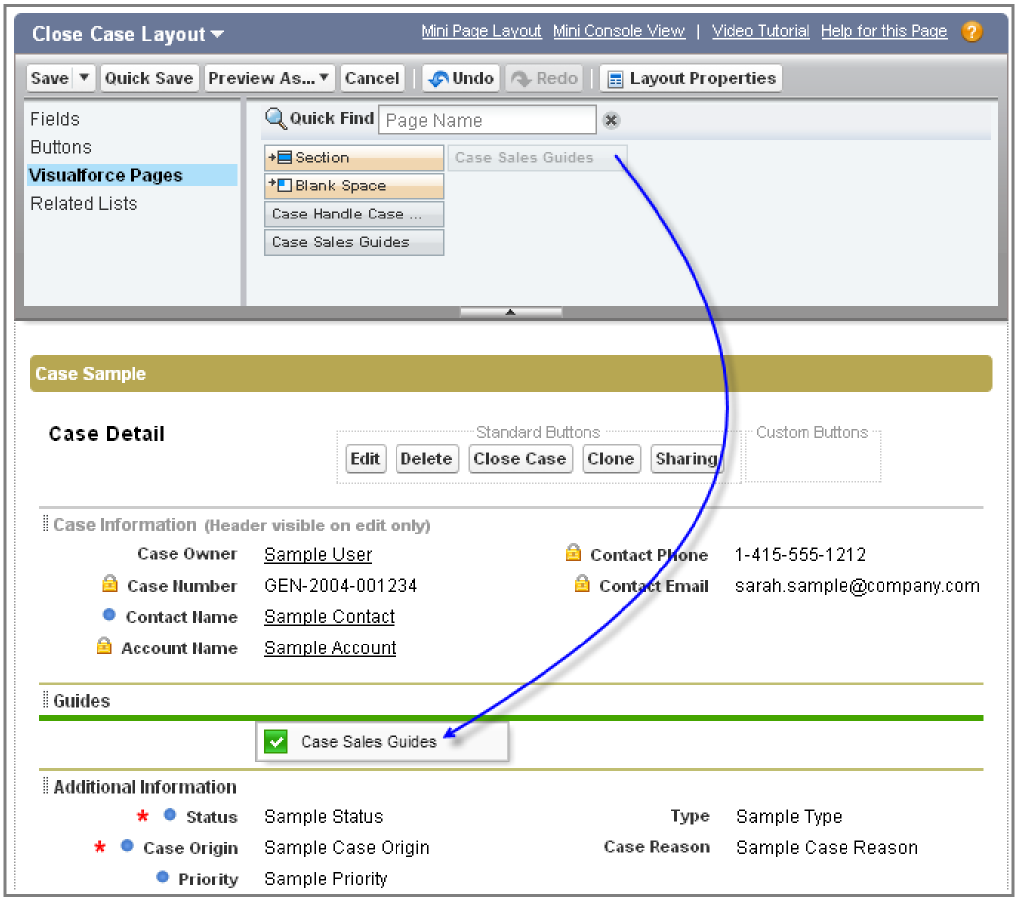
2. Select Visualforce Pages. Within the area on the right, drag Section to the place where you want the guide to appear.



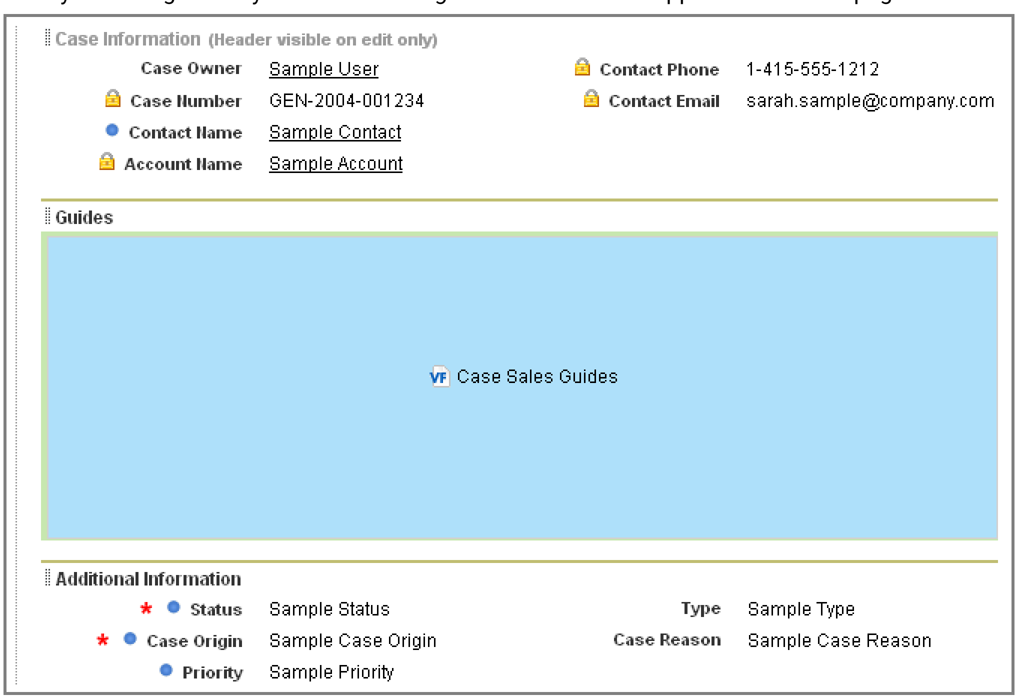
3. Set the section's attributes by filling in the displayed dialog.



4. Drag the guide to this newly created Guides area.



5. Save your changes and you're done. This *guides area* will now appear on all Case pages.



Implementing Service Steps and Search Services

This topic describes how to create service steps and search services from Salesforce Apex classes.

Note: You can download an archive file that contains a sample automated step and a sample search service.

Creating an Apex-based Service Call Step

You can create new Service Call step services using an Apex Class that implements the IAeAutomatedStepService interface.

IAeAutomatedStepService Interface

To create the new services, you first implement two interface methods:

- describe()
- invoke()

describe()

Returns AeServiceDescription, which contains metadata for services.xml.

invoke()

Invoked at runtime.

For example:

```
global class icrtStep_EchoService implements
                         IAeAutomatedStepService {
    global AeServiceDescription describe() {
          // called during synchronization
       }

       global AeServiceResponse
               invoke(AeServiceRequest aRequest) {
          // called at runtime
       }
}
```

Service Discovery with Apex Automated Steps

When you synchronize your Salesforce organization, Apex services are discovered based on their class name. If a class name starts with "icrtStep", it is recognized as a Service step (an "Apex automated step"). For example, the Apex class named "icrtStep_EchoService" is a service.

Process objects or object lists are serialized to a JSON payload format, similar to the process object input or output in other processes. They are sent to Apex where the System.JSON class in Apex can parse the JSON.

To return a process object, define the outputs as either 'reference' or 'objectlist' and serialize the objects to JSON so Process Designer converts them back to a process object.

For example, for output parameters:

```
global AeServiceDescription.AeServiceParameterDesc
       output = sDescription.addOutputParameterDesc('objOutput', 'reference');
       output.addParameterOption('referenceTo','Account');
```

Service Interface

Apex services must implement two methods:

- describe()
- invoke()

describe()

Provides information discovered for use by the cloud service.

Returns an instance of **AeServiceDescription**, which has the following fields:

Field	Description
serviceType	Type of service , either a service step or a search.
namespaceQualifier	Organization-specific namespace.
name	Service class name.
displayName	Name displayed to the user.
description	Text describing the service.
inputParameter (0..n)	List of input parameter descriptions. `(AeServiceDescription.AeParameterDesc)`
outputParameter (0..n)	List of output parameter descriptions. `(AeServiceDescription.AeParameterDesc)`

invoke()

Implements the runtime behavior of the service.

Parameter	Description
objectId	ID of the Applies To object from the host's context.
objectType	Type of the Applies To object from the host's context.
parameter (0..n)	List of AeServiceParameter name/value pairs.

Takes an instance of **AeServiceRequest** as input.

Returns **AeServiceResponse**, which has the following fields:

Field	Description
parameter (0..n)	List of AeServiceParameter name/value pairs.
errorInfo.AeErrorInfo	Conveys error information. Use this to relay error information instead of throwing an uncaught fault within your service
errorInfo.errorMessage	Error message.
errorInfo.invalidData	Field-level error details.

Custom Search Services

You can implement search services as Apex classes. Search services use the same interface and discovery mechanisms as step services, with the following differences:

- Set the service type to search in AeServiceDescription:
  `AeServiceDescription.serviceType = 'search';`
- There should be one output parameter, Search Results, that returns the data.
- To return a valuelist, the invoke() method should return name/value pairs representing the name/id of the objects returned.

The following example returns a valuelist of account references filtered by OwnerId:

```
List<Account> accountList =
    [SELECT Name, id FROM Account WHERE OwnerId =:userRef LIMIT 20];
      for (Account account: accountList)
      {
         // For a search service, return name/id as parameters
         response.addParameter(account.Name, account.Id);
      }
```

Republish the Salesforce Connection

To make custom automated step and search services visible, you need to republish the Salesforce connection from Application Integration.

Note: You need to republish the Salesforce connection only when you make a change to the class interface, not when you make a change to the class.

CHAPTER 5

Mobile Guides

This chapter includes the following topics:

Mobile Guides Overview

The procedures you use when you create guides that run on smart phones are identical to those you use when you create guides that run on a PC or a Mac.

However, the smaller screen affects your design in the following ways:

- Mobile forms typically have just a few fields, with more steps, due to the limited display area.
- The order of fields in a mobile guide is critical because of the limited number of visible items at any one time.
- Use of images is more limited due to size constraints.
- Guides are often launched from URLs that are provided to or are created by your users.
- Keep in mind that users will use the phone's voice processing software.
- Typing is less convenient on a mobile device. Where possible, let the user choose data from lists or checkboxes.
- You can design a guide with the Run On setting as Desktop/Tablet and change the setting to Smart Phone when the design is complete.
- If you need a guide to run on both a mobile phone and on a larger device, you can do much of the work in embedded steps, to make it easier for use on both device types.
- Guide Designer requires that all guides, even embedded guides, have a Run on setting. However, at runtime, it uses the Run on value set for the embedding task and ignores the Run on setting in the embedded guide..
- Depending on the type of phone and operating system, users may have limited abilities to format text.

You can run a guide in your Android phone or iPhone, using the phone's browser.

Note: Guides display and run only on the device type selected for the guide (Smart Phone or Desktop/Tablet).

Smart Phone Setup

This topic explains how guide users can install an icon that launches a guide in the phone's browser.

Using Wizards on an iPhone

On an iPhone, users must:

1. Enable cookies for Guide Designer. If they are not set, go to iPhone Settings and set **Safari > Accept Cookies** to **Always**.
2. Open Safari and login to Salesforce. In most cases, just go to http://login.salesforce.com/.
3. After logging in, select the *Home* tab if it isn't already selected. Although it may be small, you will see the Guide Designer Wizards area.
4. Tap the **New Window** button in the Guide Designer Wizards area. A list of guides appears in its own window, making it much easier to read.
5. Tap the iPhone icon that lets you bookmark the page or add it to the Home Screen.

Using Wizards on an Android Phone

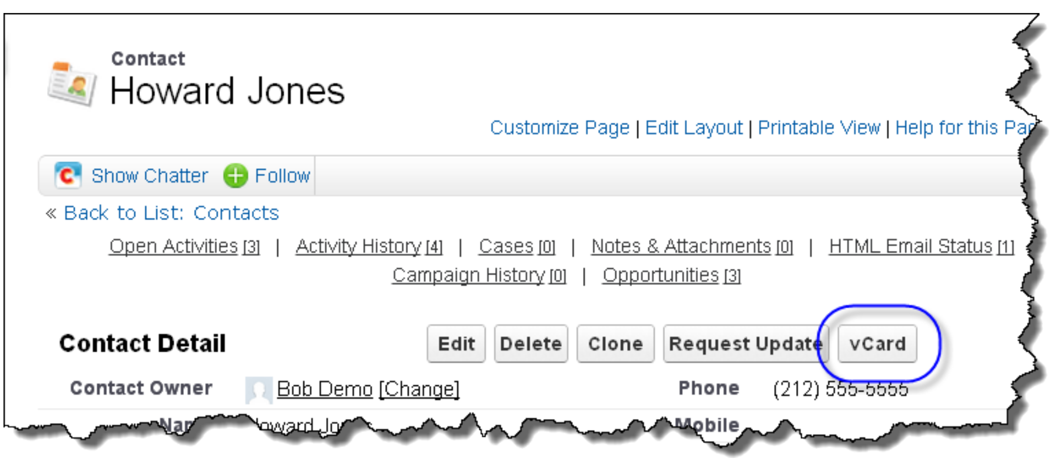
On an Android phone, users must:

1. Open their browser and login to Salesforce. In most cases, just go to http://login.salesforce.com/**END**.
2. After logging in, select the *Home* tab if it isn't already selected. Although it may be small, you will see the Guide Designer Wizards area.
3. Tap the **New Window** button in the Guide Designer Wizards area. A list of guides appears in its own window, making it much easier to read.
4. Touch the star at the right of the URL bar, perhaps edit the page's name, tap **Bookmarks** or a folder in the **Add To** menu, and then tap **OK**.

For more information, see documentation at the Google support site.

Contacts on Your Cell Phone

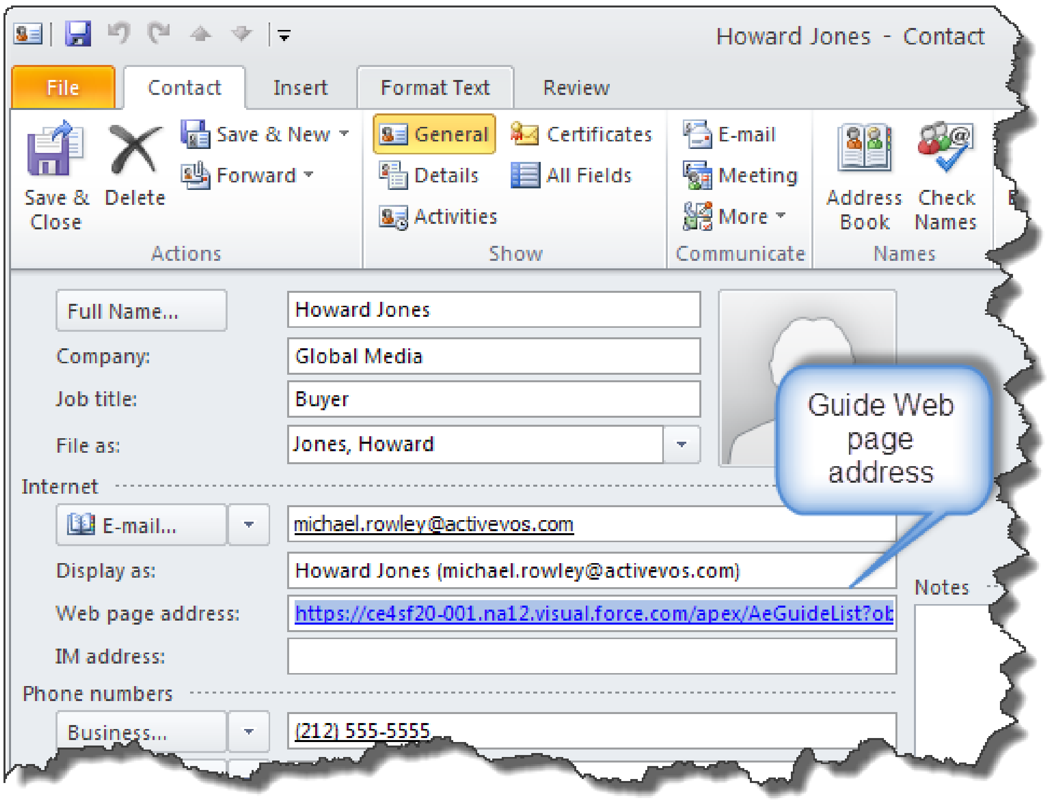
The mobile phone setup wizard adds a **vCard** button to your Contact pages.



Pressing this button tells Salesforce to create a **vCard** file (it has a `.vcf` extension), and sent it to your browser. For most browsers, click on the file's name to add a contact to your personal contacts.

- **Firefox**: The name is in its **Download** dialog.
- **Chrome**: The name is at the bottom within the window's frame.
- **Internet Explorer**: You can run it from the displayed dialog box.

After it loads, you can see it in your contact manager. Here, for example, is the contact within Microsoft Outlook. Notice the Web page address link:



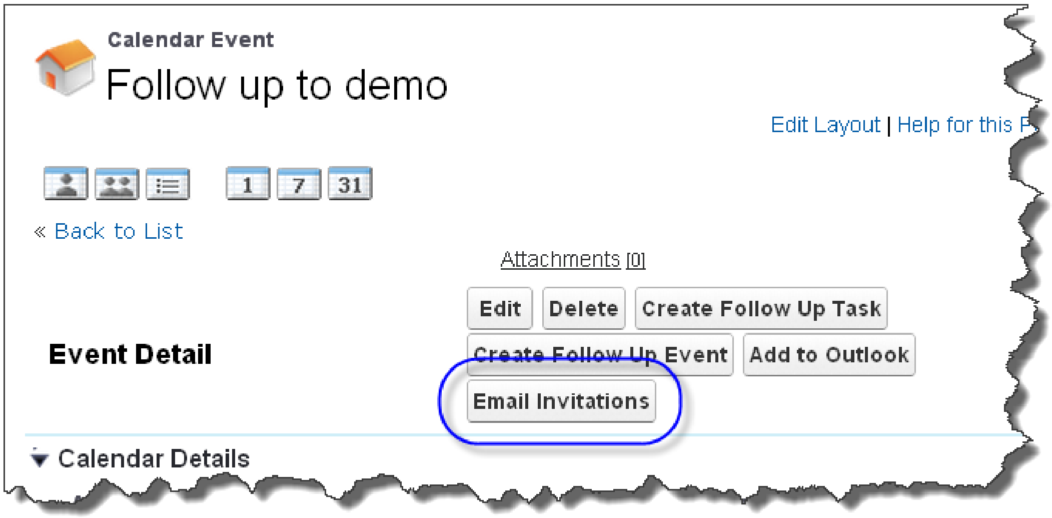
This link also appears in this contact on your smart phone after it syncs with your contact manager, which is the left phone in the next figure. Clicking on the link opens your phone's browser and it displays the guides that you can use for this contact. You can now click on a guide's link to open the guide's first step. If there is only one guide, this guide displays.

Note: Only mobile guides are displayed when you use this link. Guides that were designed for display in your PC's browser are not shown. Similarly, clicking on this link while using a contact manager such as Outlook when you are using a PC or a Mac tells Guide Designer that it should only display non-mobile guides.

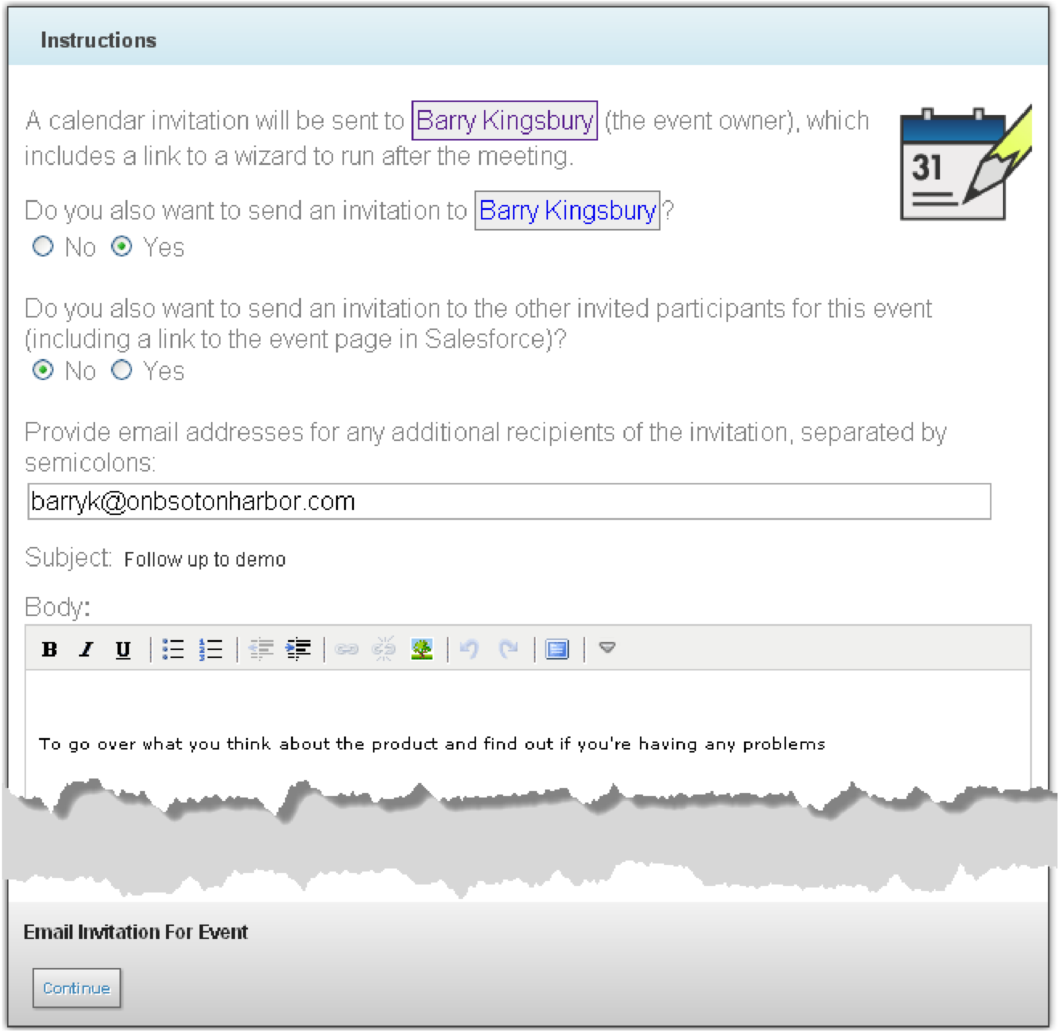
Smart Phone Calendar Entry

When smart phone support was added to Guide Designer, the setup wizard added an **Email Invitations** button to Salesforce Calendar Event pages.

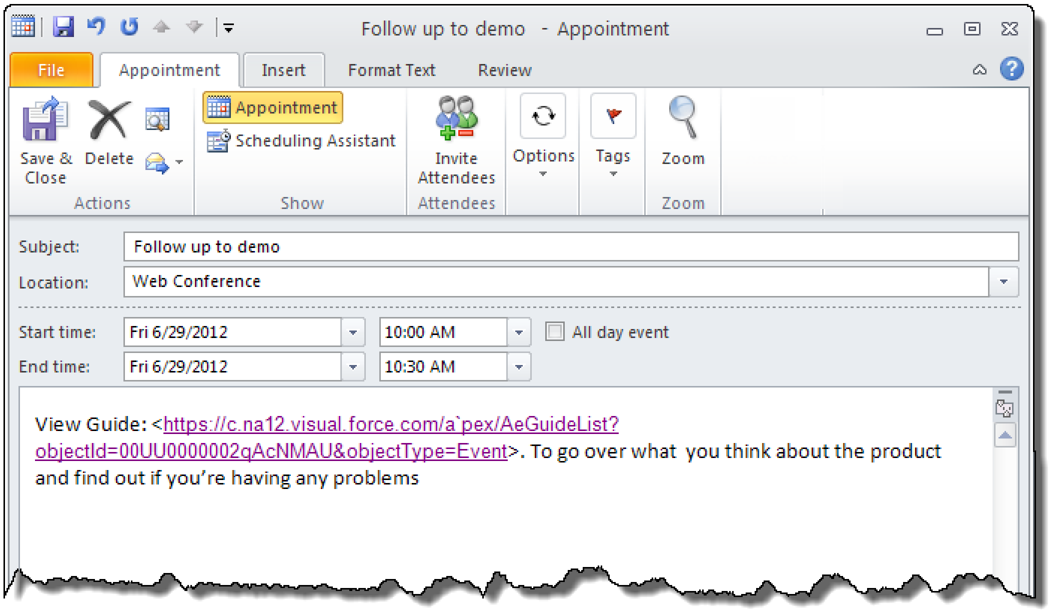
Note: This button only appears on the page that Salesforce displays when it shows you information about already created events. It does not appear on the page containing the form that you use to create the event.



After you select this button, the wizard displays a wizard that creates iCal files. After the completing the wizard, Guide Designer sends mail to the people you've identified. For many, it will attach an iCal file. It actually sends two different iCal files. The first places the event on a calendar after it is executes. It also has a link to the guide that the event's host will use. The second sends iCal files to the people attending the event and who are identified in your Salesforce objects. This iCal file just places the event on a calendar. It also has any text that you added. If the person is not identified in your Salesforce objects, the wizard just sends the text you entered.



After making choices in this step and perhaps adding additional recipients, click the **Continue** button to send invitations. The wizard attaches an iCalendar ics file to the email. When the recipient clicks on it within the received email, the event is added to the recipient's calendar. The event owner receives a slightly different email as it also has a link to the guide navigator. Here is a Microsoft Outlook example containing the link:



On an iPhone, you will receive the Event data shown in the left screen in the next figure. After tapping this link, the wizard sends the first step to your phone, which is shown in the right screen.

Importing iCal files on Android Phones

Android phones, by default, do not support iCal ics files. Instead, Android relies on apps that you can download into your app using Google Play. Informatica does not validate or test any of these apps. Our advice with regard to choosing an app is to choose one with many downloads. Also, it is possible that some apps do not work with the Android version that you are using.

In testing apps, attachments within gmail were *usually* opened without needing support from another app.

In most cases, the app works by recognizing the ics file type used by iCal files. When you click on an iCal email attachment from another app, you are usually asked if you want to download or preview the attachment. One of the options should be the name of the app you installed that is associated with iCal files. You should also see a checkbox that lets you set this app as the default action. If you have more than one calendar app on your phone, you will need to choose which calendar the iCal app will use. Finally, some apps import the iCal file and leave the calendar entry in an open state so that you can view and modify it. It is not saved unless you explicitly save it. Other apps just save it and you will have to open it separately. Others save it and leave it open.

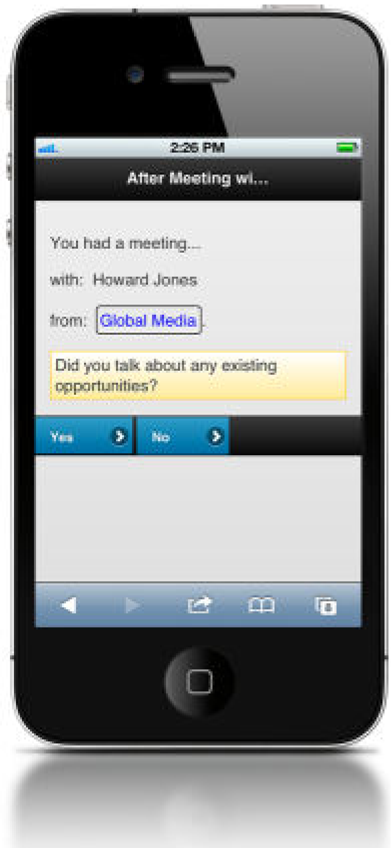
CHAPTER 6

Meeting Follow-up Wizards

This chapter includes the following topics:

Meeting Follow-up Wizards Overview

The meeting follow-up wizards, built with Guide Designer for Salesforce, accelerate and automate many routine tasks that are performed after meetings. These include recording notes, creating contacts, and assigning follow-on tasks. The Guide Designer wizards add additional integration between Salesforce and your smart phone.



After the wizards are installed within your Salesforce pages, you are ready to run them. Two of the wizards run from within your mobile phone. The third runs from a Salesforce page.

Running Wizards on your Smart Phone

Two of the follow-up wizard discussions present guides that run on mobile phone. A Guide Designer Wizard is not an app; that is, you do not have to download something from iTunes or Google Play. Instead, the wizards are web applications that run in your smart phone's browser.

As you will see, running a wizard on your smart phone lets you add information about a contact, opportunity, lead, or any other of your Salesforce objects much closer to the time when you met with or talked to someone. You do not have to start your laptop and perhaps find a Wi-Fi hotspot. Instead, you can interact with Salesforce using a wizard that helps you enter information and does it in a way that fits into the small size of your smart phone's screen.

When wizards are defined, they are defined in a way that makes each step (that is, each screen) appear in a way best suited for your phone. If your site has created wizards, you can go display just your wizards. These wizards usually let you create new objects. For example, a guide could help you create a new lead.

When you login into Salesforce, it displays your Salesforce home page. If you scroll down, you will see a Guide Designer Wizards section. This section contains the wizards that you can execute from your home page. Directly underneath the list of Wizards, you will find a New Window button. Tapping on this button displays the wizards on their own page. You may want save that page as an icon on your phone's home screen.

Contact Wizards

After talking with a contact, you can update Salesforce in two ways. If you are using Salesforce on a PC or Mac, use the Sales Call Follow-up wizard to enter new tasks, notes, contacts, and opportunities. You can also set follow-up tasks for yourself and others. To run this wizard, go to the contact's page in Salesforce and select this guide from the Guides section. (The Guides section was added to your contact pages when your administrator installed the wizards.) For more information, see "Running a Wizard from the Contact Page" on page 71. If you are using a smart phone, you can use the "After Meeting Wizard" on page 68.

The Mobile Phone Setup Wizard added a **vCard** button to your Salesforce contact pages. After pressing this button, the Guide Designer wizard creates a **vCard** file that your browser can download. Opening this file adds a contact to your contact manager (for example, Microsoft Outlook). It will also add a link that, when clicked, displays the guides that you can use for this contact. For more information, see "Contacts on Your Cell Phone" on page 59.

Smart Phone Calendar Entry

The Mobile Phone Setup Wizard add an **Email Invitations** button to your Salesforce Calendar Event pages. After pressing this button, a wizard displays a page for customizing an invitation and perhaps sending it to more people. The invitation contains an iCalendar file, which when the recipient clicks on it, adds the event to the recipient's calendar.

For more information, see "Smart Phone Calendar Entry" on page 61.

After Meeting Wizard

The After Meeting Wizard lets you accelerate and automate many of the routine tasks that are performed after meetings. It lets you update existing opportunity information and add new ones after talking to a contact. It also lets you add new contacts, notes, and create follow-up tasks.

Select a heading to see the different operations that the Wizard guides you through.

Identify Contact and Update Opportunities

Step 1

If the guide did not start with a known contact, who did you contact?

Step 2

If the meeting was associated with an opportunity, then update the stage, amount, close date and next step for the opportunity. Also add meeting notes which will be added to the activity history for this opportunity.

Note: When entering text such as notes, it is far easier to use your smart phone's voice recording capabilities if it has them.

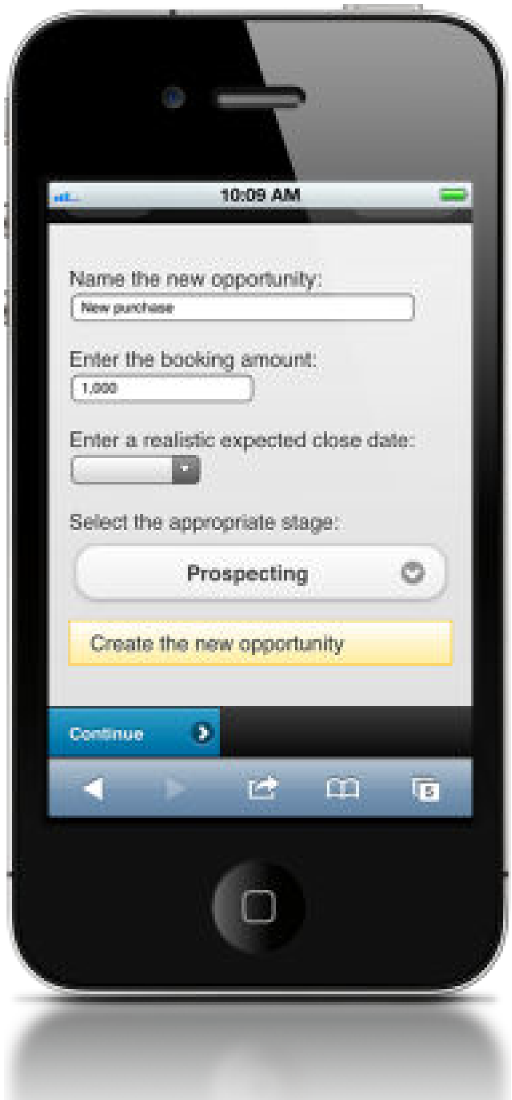
Step 3

Provide (or update) the role that this contact plays in this opportunity (e.g., business user or decision maker).

Additional or New Opportunities

Step 4

The wizard then presents a list of all other opportunities for this account. Select any that were discussed in this meeting. The wizard then walks through each selected opportunity and allows you to update them.



Step 5

Add any new opportunities that were discussed, providing key fields for each.

New Contacts

Step 6

Create contacts for any people from the meeting that were not already contacts.

Follow-Up Tasks

Step 7

Create a follow-up task for yourself.

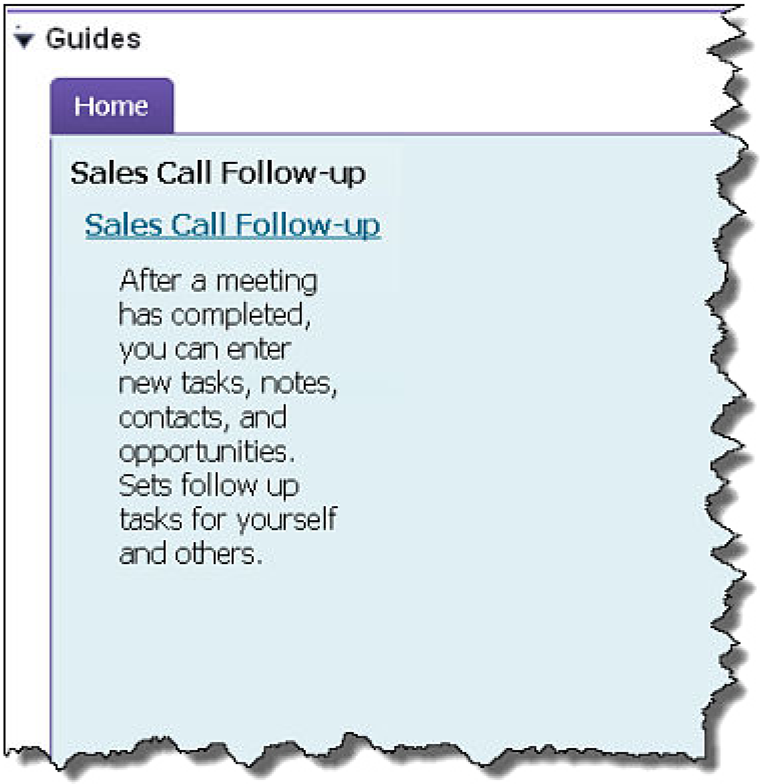
Step 8

Create follow-up tasks for other people.

Done!

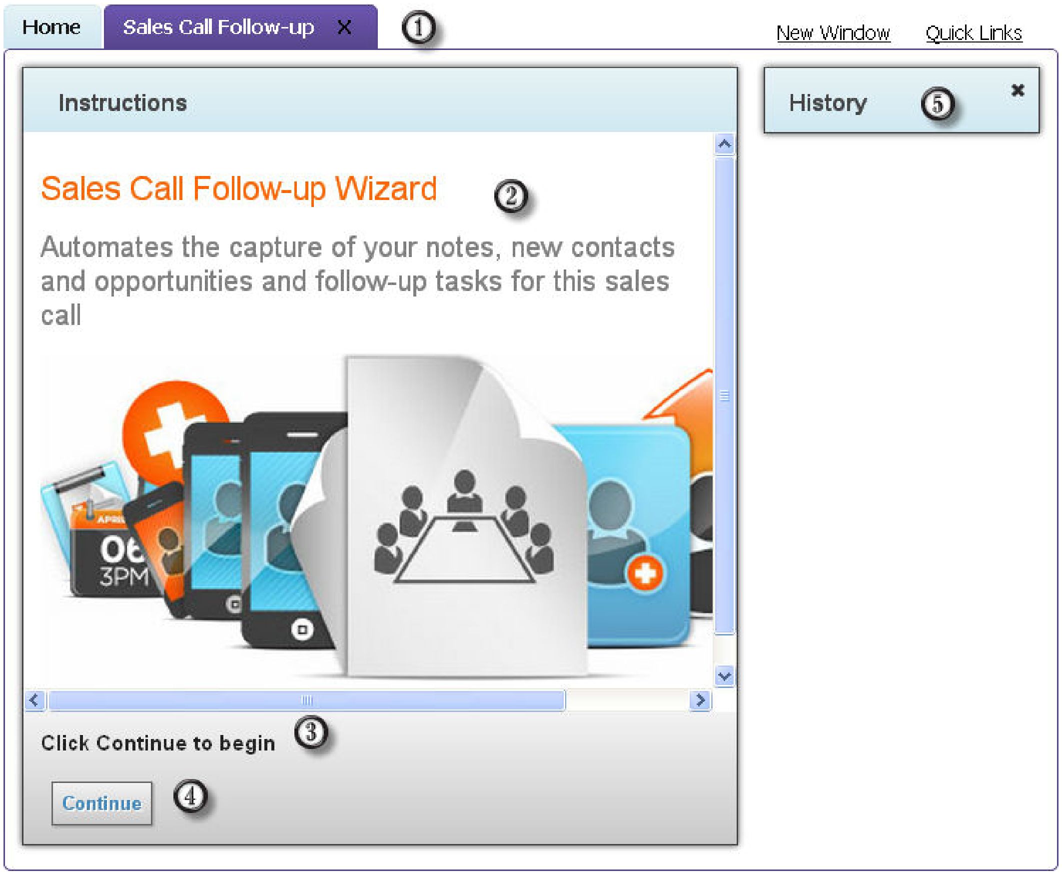
Running a Wizard from the Contact Page

After your administrator installs the meeting wizards, all of your contacts will have a **Guides** area. This area contains the "Sales Call Follow-up Wizard." This wizard only displays when you are using a PC or a Mac.



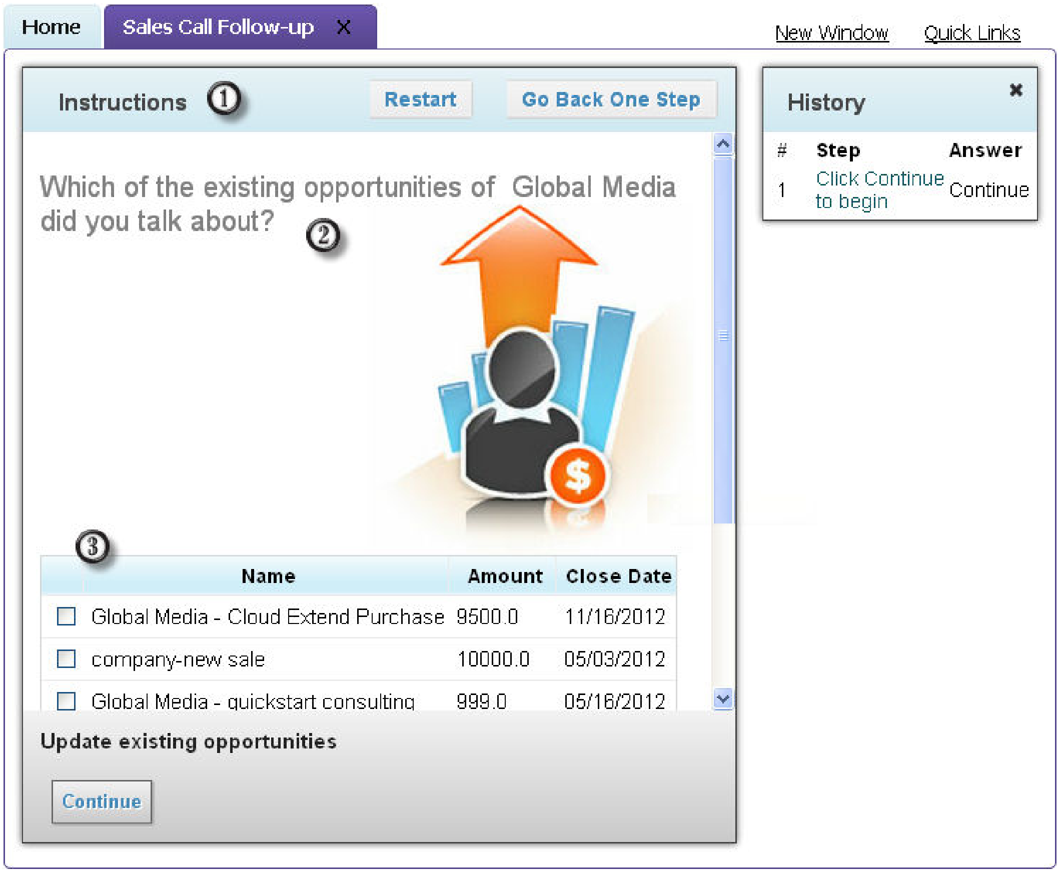
Click on the "Sales Call Follow-up" link to start the wizard. The first step within the wizard displays. (All screens within the wizard are called "steps".) The table following this figure explains what is within this first step.

The way information is presented from step-to-step may differ slightly; however, all are similar.



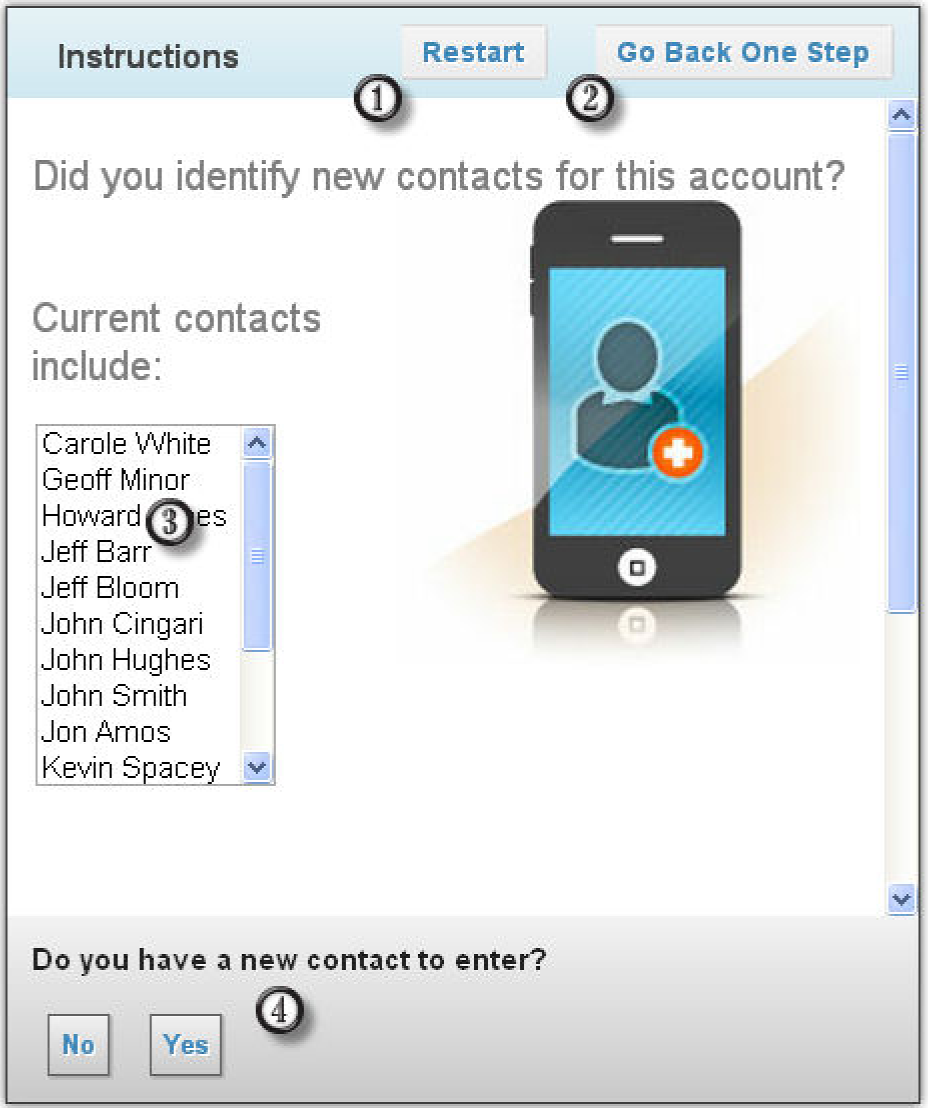
①	A tab for your wizard and others wizards that you are using.
②	The step's main area. This area can contain instructions, information, areas into which you will be able to type text, select check boxes, and so on. The step shown here only has text as it describes what the wizard does.
③	A prompt area. This text tells you which button to press so to select the step that the wizard next executes. This step has only one button.
④	Pressing a button in this area tells the wizard what the next step will be. In many cases, a step has more buttons than are shown here. Each button tells the wizard what the next step will be. For example, a step that asks a question might have Yes and No buttons, and the step that follows depends upon which you select.
⑤	The history of the steps you've executed. A new item is added after you press a button within the prompt area. Each step in the history is actually a link. Clicking on it reverts the wizard back to a previously executed step. This also undoes any of the changes made to Salesforce since the step you're going back to executed. This could, for example, delete a lead that another step created. Note that you can close the history window to make more room for the main wizard window by clicking on the "x".

In this example, press the Continue button after you've read the step's text. The wizard now displays the following step.



①	The "Instructions" label is in all steps.
②	A question. Most steps begin with a question or instructions. A form into which you'll enter information usually follows.
③	Here, the wizard displays all of the opportunities with this company. The wizard uses the contact's company name to locate and display this Opportunity information.

After advancing a few steps, the wizard can display:



This figure contains two new elements added to the step. Also, two others are slightly different.

①	The Restart button tells the wizard to return to its first step. Unlike using the history to go back to the first step, clicking restarts will not undo any of the changes made by the wizard (such as deleting objects).
②	While you can return the wizard to any of its previously executed steps by clicking on the step in the History list, it is often more convenient to press the Go Back One Step button if all you want to do is return back to the step just executed.
③	This table, automatically generated by Guide Designer, contains a list of your contacts for this account.
④	Unlike the two previous steps shown in this topic, this step contains two buttons. Other steps can contain more (or perhaps many more).

CHAPTER 7

Controlling Who Sees Guides

This chapter includes the following topics:

Controlling Who Sees Guides Overview

Two ways exist for you to control which of your users can see guide navigator areas on Salesforce pages. The most common is to uses page layouts that are assigned to user profiles. If you need more granularity than this method provides, you can use permission sets. A permission set allows you to assign an object to a set, then assign that permission to selected users. (For more information on permission sets, see Permission Set Overview in the Salesforce documentation.)

Using permission sets, all users in a profile will still see a page layout that has a section for guides. However, only users associated with the "Guide User" permission set will actually see the guide navigator. Other users that open the "Guides" section of their page will see a message something like:

```
Content cannot be displayed: You do not have sufficient
privileges to access the page: /apex/ce4sf20_001__AeCaseSalesGuides"
```

Using permission sets is a three-step process.

1. Go to the Visualforce page for guides and change their security setting so not users are allowed. Do this here because Visualforce pages for guides are typically named "*objectName* Sales Guides". Now, no one is able to see the guide navigator.
2. Create a permission set called "Guide User". In this permission set, click on "Visualforce Page Access" and give access to all of the CE Visualforce pages.
3. Go to the users that should have access to Guide Designer and associate them with that permission set.

Now, only these users will see the guide navigator.

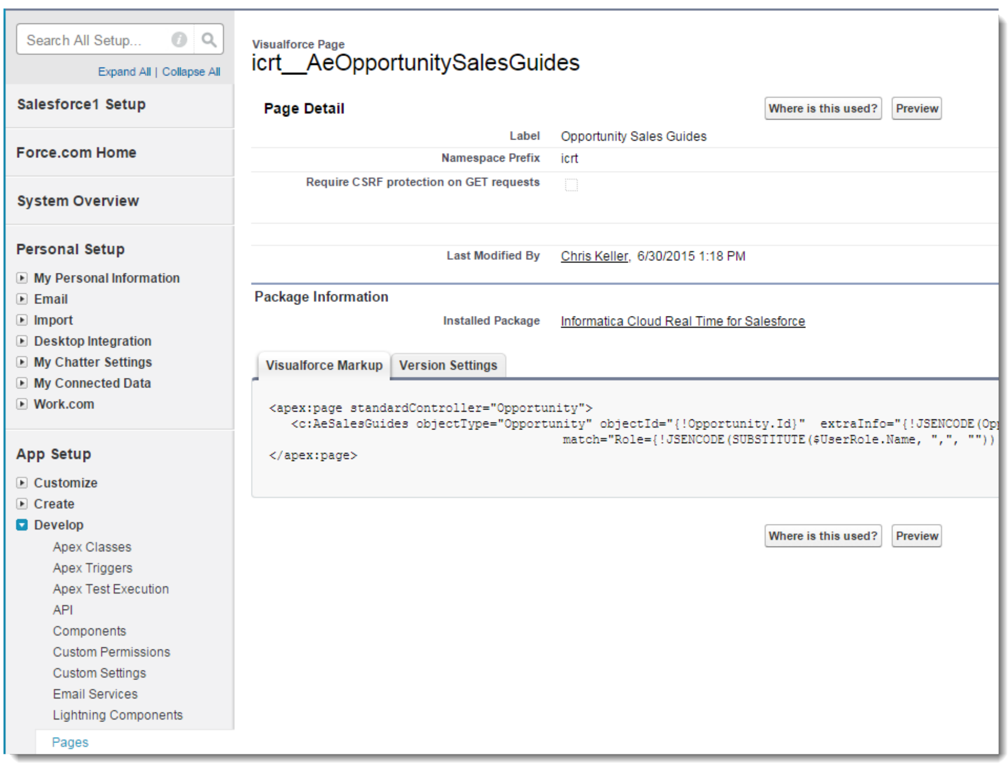
Filtering Using Tags

When you define a guide's properties, you specify the object to which the guide applies and all users who have access to the object can see the related guides.

You can set conditions under which guides appear by Using the **Tags** area on the **Guide Properties** dialog. For example, if you might want to display a certain guide only for leads with an "Open - Not Contacted" status. In this case, you first create a tag for the guide that is used as a display filter. In addition to the built-in capability filters for some object fields, you can then customize guides to apply these tags.

Note: You can also simply set Guide Properties to determine where guides display. For details, see "Show Guide Only When" on page 87.

To filter from the Visualforce page, use this approach:

1. From the Salesforce Setup menu, select **App Setup** > **Develop** > **Pages**.
2. Select the Visualforce Markup tab:
3. 

4. Add a filter to this page to create a tag, using the guidelines below. For example, a guide could filter by account types, account industries, or user roles.

Guidelines

Suppose you want users with different roles to see and be able run different sets of guides for Opportunities. To do so, simply create a tag like `Role=CEO` on your guide. Since the Opportunity object type is already set up to filter based on role, this guide will now only appear if the user's role is CEO.

The equals sign (=) determines how the guide use the tag's contents for matching. The guide will be displayed only if the object matches all of the guide's tags.

Notice this condition in the Visualforce page definition:

```
match="Role={!JSENCODE(SUBSTITUTE($UserRole.Name, ",", ""))}
```

When the Opportunity page loads, because `$UserRole.Name` is a match (in this case,`match="Role=CEO"`, the guide displays. (You can ignore the JSENCODE and SUBSTITUTE functions.)

If a guide does not have a tag with an equal sign, it is always shown, you can define match criteria to excludes a specific value, or hide a guide by setting a match definition will never occur, for example, `Role=foo`.

In the Account page definition (AeAccountSalesGuide), there are three possible matches. Using the same kind of logic, you could create a guide that matches on any of the fields that are on the page:

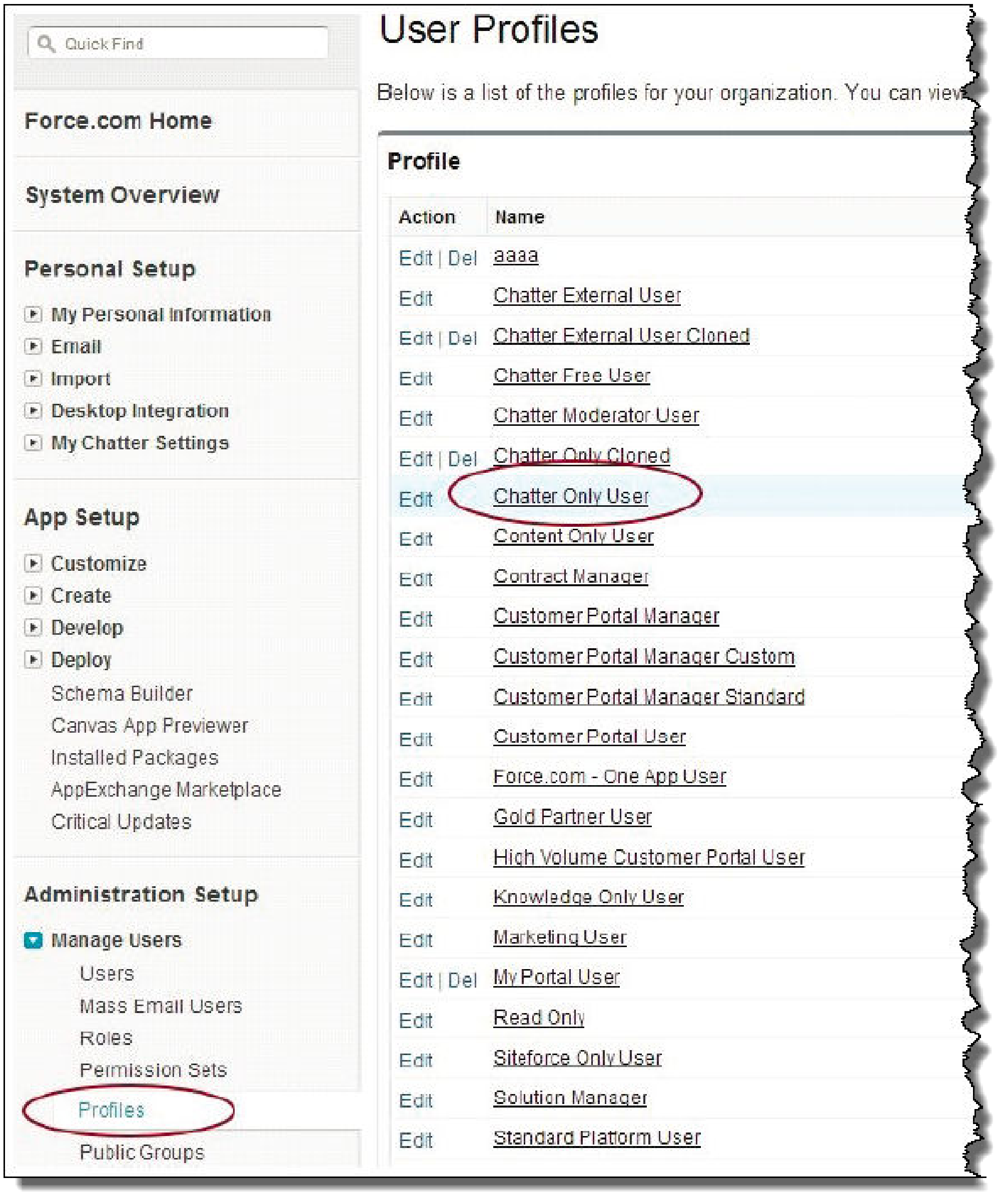
```
<apex:page standardController="Account">
     <c:AeSalesGuides objectType="Account" bjectId="{!Account.Id}"
          extraInfo="{!JSENCODE(Account.Name)}"
          match="Role={!JSENCODE(SUBSTITUTE($UserRole.Name, ",", ""))},
          Type={!JSENCODE(SUBSTITUTE(Account.Type, ",", ""))},
          Industry={!JSENCODE(SUBSTITUTE(Account.Industry, ",", ""))}"/>
</apex:page>
```

Enabling Guide Designer-Related Classes for New Profiles

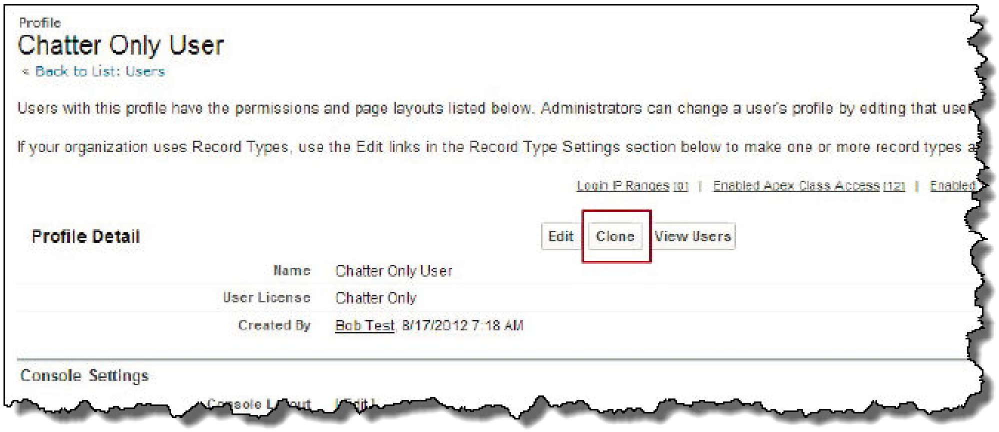
This topic presents a set of operations that let you enable Guide Designer-related classes for new profiles within Salesforce and then enable the Guide Designer Apex-related classes and the Visual Force pages for the profile.

As an example, this topic shows how you launch a guide with Chatter Only License.

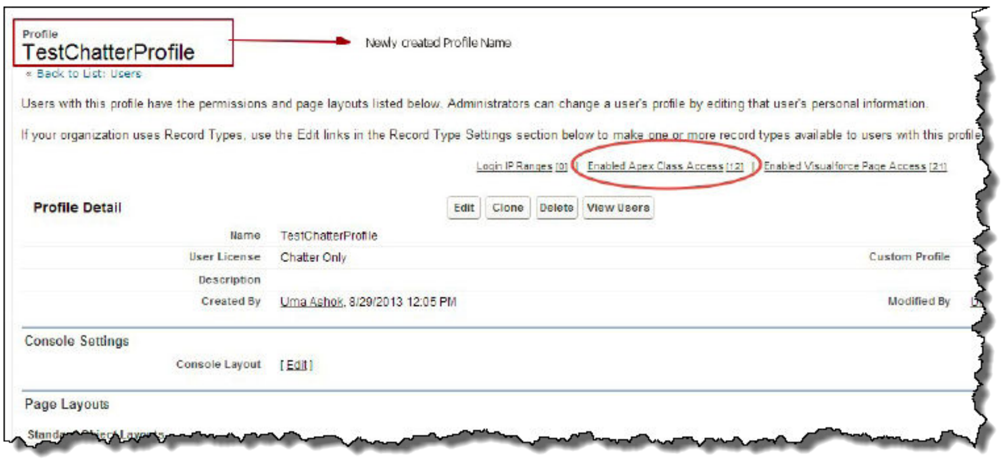
1. From the list of User Profiles (located within **Setup > Administration Setup > Manage Users > Profile**), click the link of the Chatter Only User. (Do not select Edit.)



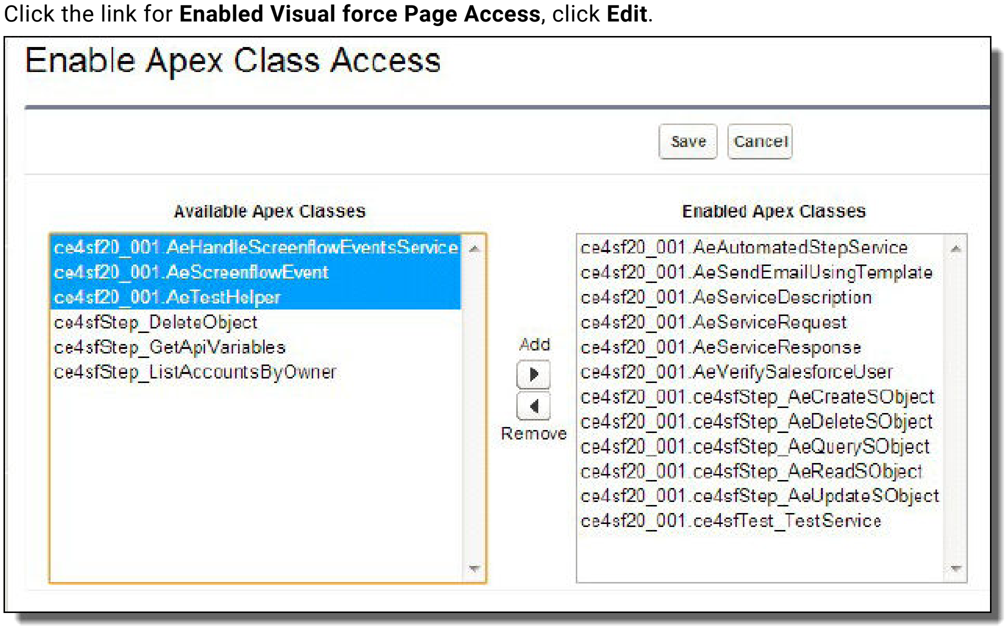
2. Click the **Clone** button, specify a profile name, and then click **Save**.



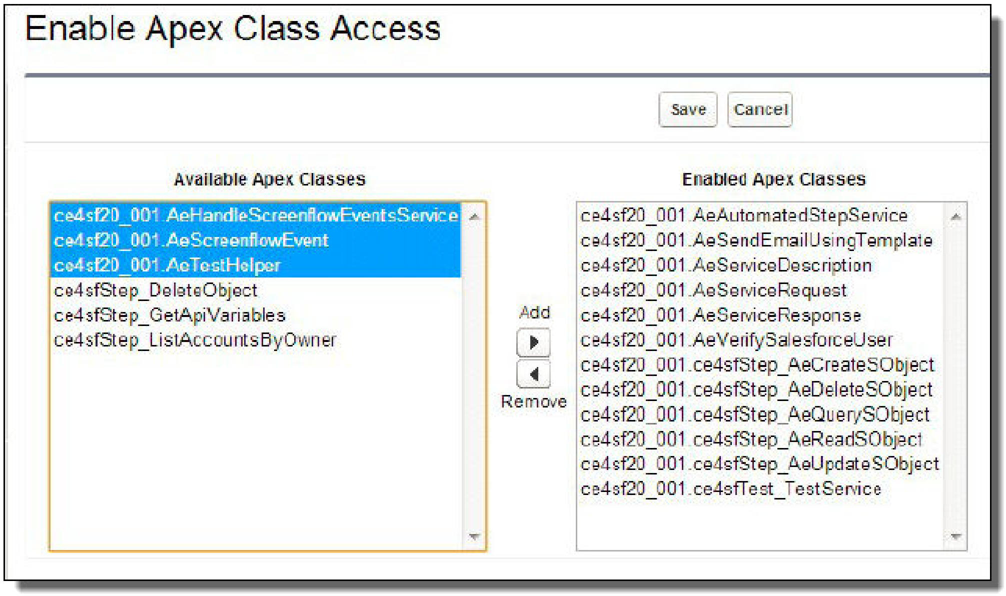
3. Click the link for **Enabled Apex Class Access**, click **Edit** and select all classes starting with "ce4sf20_001 from the Available Apex Classes and add them to Enabled Apex Class. Click **Save**.



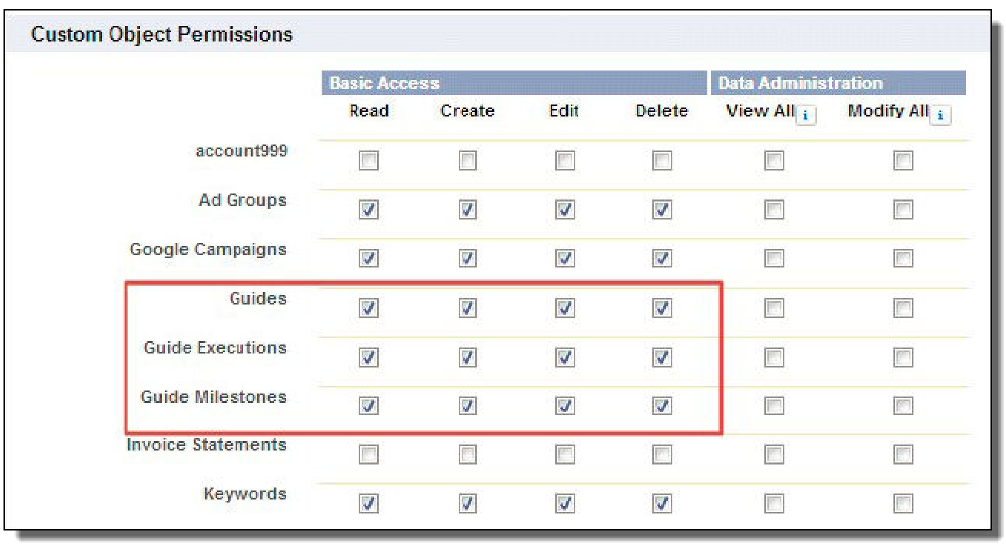
4. Click the link for **Enabled Visual force Page Access**, click **Edit**.



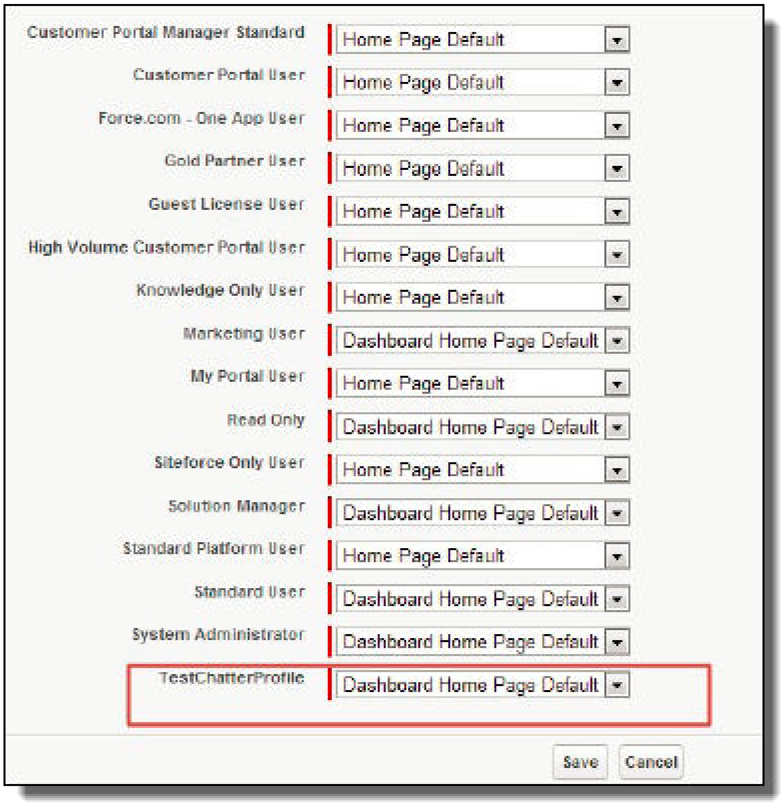
5. Select all pages starting with "ce4sf20_001" from the Available Visual force Pages and add them to Enabled Visual force Pages and click **Save**.



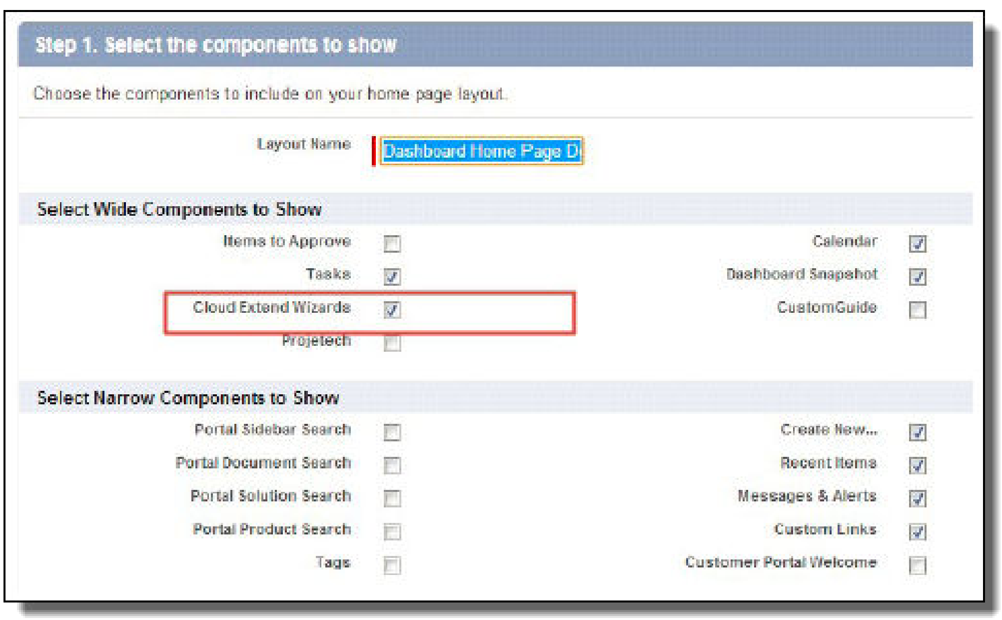
6. Edit the profile, navigate to Custom Object Permissions, and set the Access rights for the custom objects for Read, Create, Edit, Delete as appropriate. Refer to the following snapshot below for an example. Your choices will be what is required for your organization. Click the **Save** button located at the bottom of the page.



7. Go to **App Setup > Customize > Home Page Layouts**, click **Page Layout Assignment** and select a Home Page Layout for the newly created cloned Chatter User Profile. Below is an example.



8. Edit the Home Page Layout selected in the previous step and make sure that the Guide Designer Wizards checkbox is selected, click **Next**, then **Save**.



More Information

Chatter only licenses are also known as Chatter Plus. They are designed for Unlimited, Enterprise, and Professional Edition users that do not have Salesforce licenses but need access to some Salesforce objects in addition to Chatter. These users can access standard Chatter people, profiles, groups, and files, plus they can:

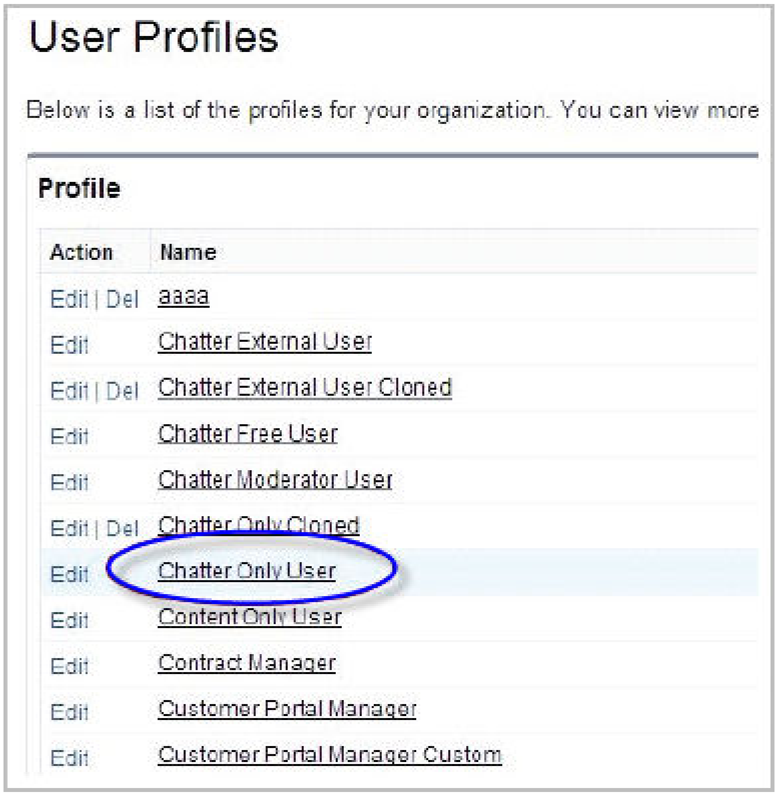
- View Salesforce accounts and contacts.
- Use Salesforce CRM Content, Ideas, and Answers.
- Modify up to ten custom objects.

Associating Guide Designer Apex Classes With a User Profile

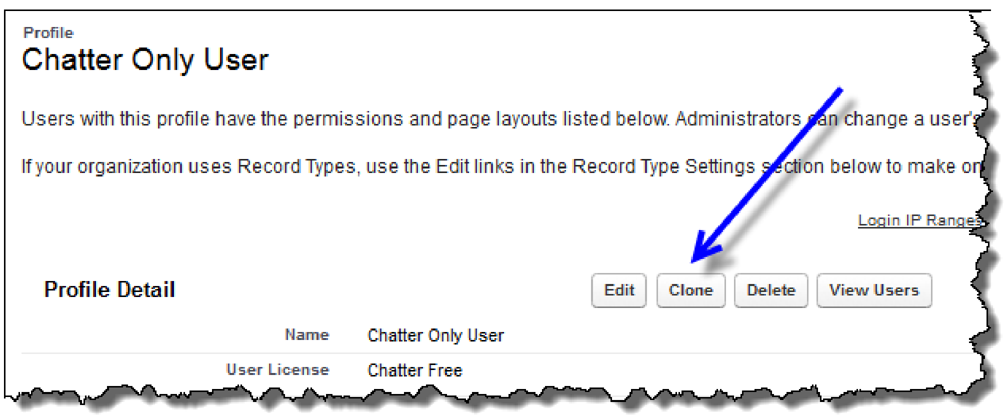
This topic shows how to create a profile and associate Guide Designer-related Apex classes to it. The procedure that you will read allows guides to run with Salesforce Chatter Only licenses.

1. Select **Setup > Administration Setup > Manage Users > Profile**.

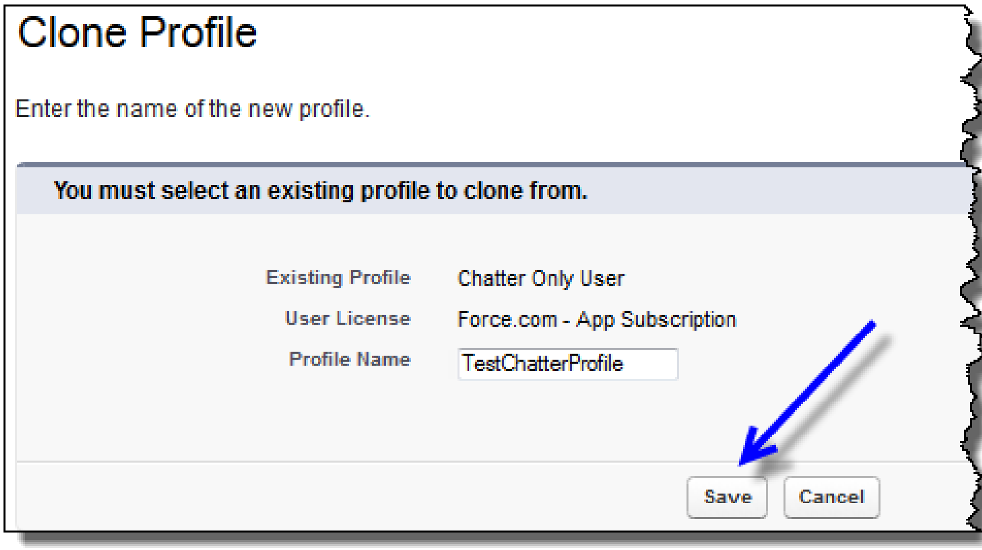
2. Select a User Profile. Here "Chatter Only User" will be selected.



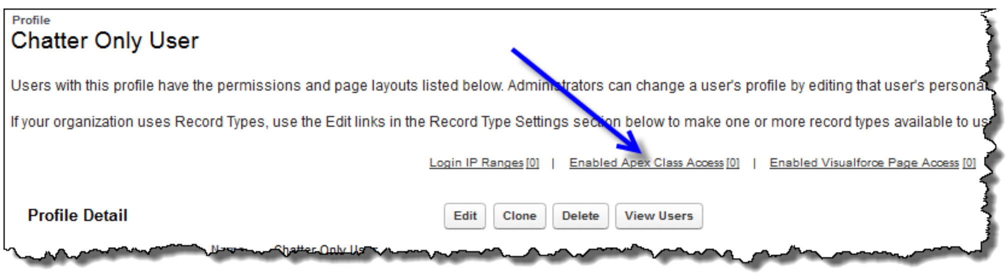
3. On the displayed page, click the **Clone** button.



4. On the displayed page, give the profile a name; for example, "TestChatterProfile" and then click **Save**.



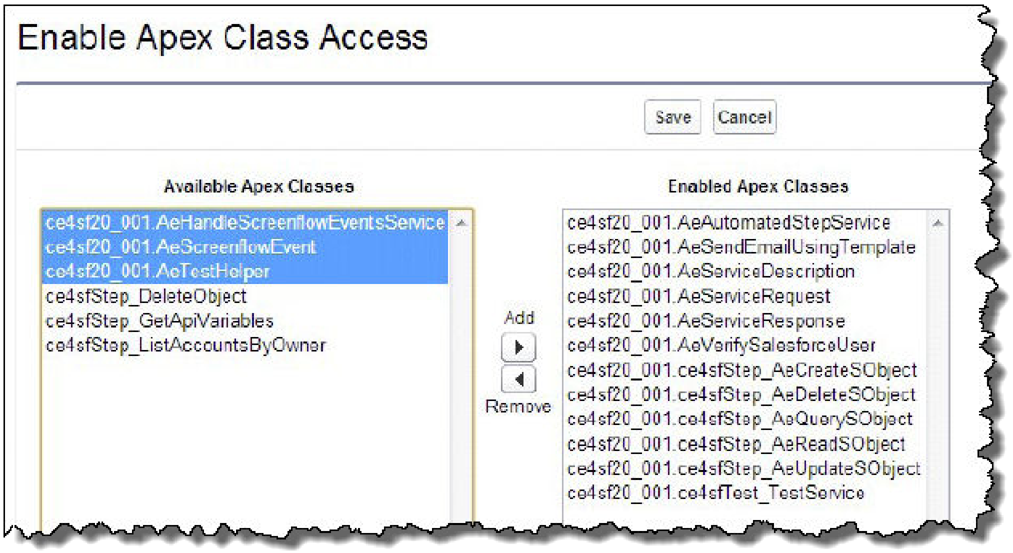
5. Salesforce displays a Profile page for the profile you just created. On that page, press **Enabled Apex Class Access**.



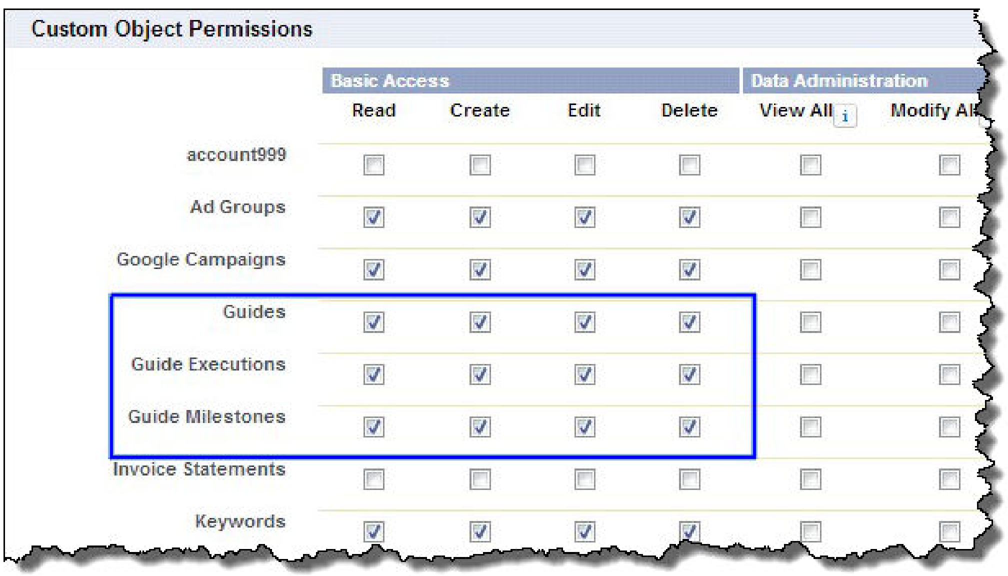
Salesforce scrolls this page and displays this area. The area may already have some classes enabled or, as in this example, none are yet enabled.

Click the **Edit** button.

6. Within the Available Apex Classes area on the left, select all of the classes starting with "ce4sf20_001" and press the **Add** button. This name may be different in some releases. However, they will always begin with "ce4sf".

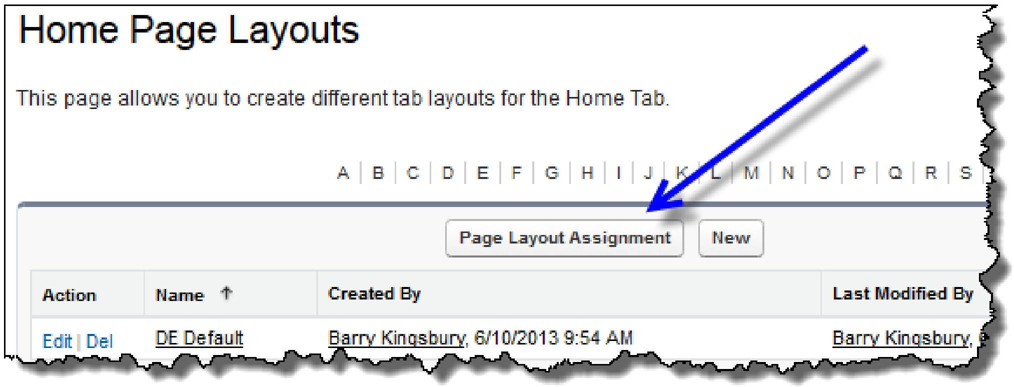
7. 

8. Scroll the profile window up until you see the Custom Object Permissions area. Set the access rights for the custom objects for Read, Create, Edit, or Delete as appropriate. Here is an example:



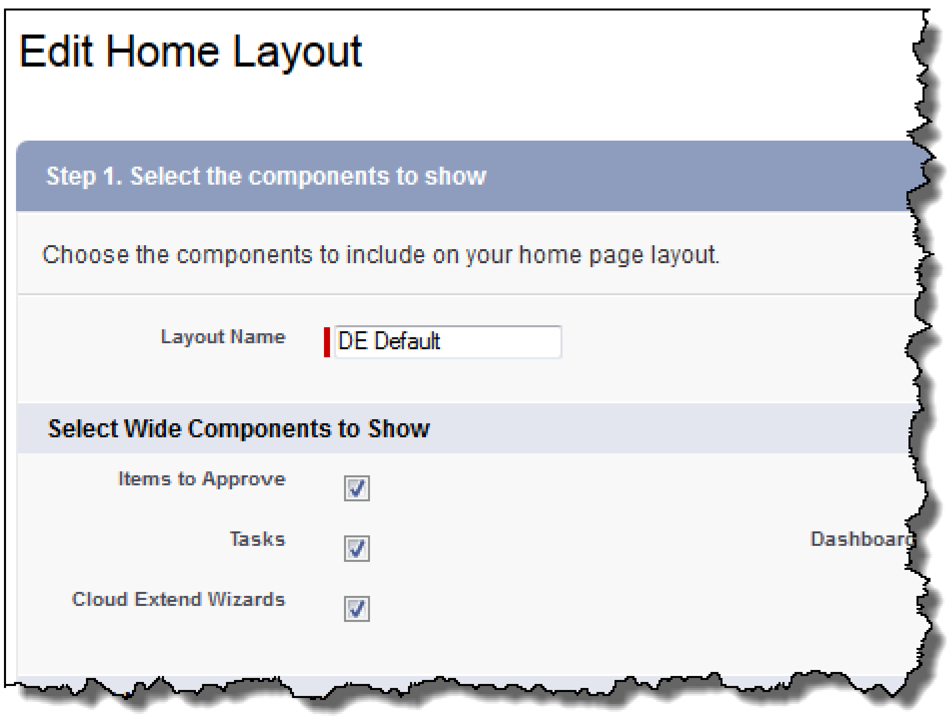
9. Click the **Save** button located at the bottom of the page.

10. Go to **App Setup > Customize > Home > Home Page Layouts**, and click the **Page Layout Assignment**

11. 

12. Select a home page layout for the newly created profile; in this example, it is "TestChatterProfile." Click **Save**.

13. Edit the home page layout selected in the previous step and make sure that the "Guide Designer Wizards" checkbox is selected. Click **Next**, then click **Save**.



Show Guide Only When

Note: This feature only applies to guides that execute within Salesforce.

You can determine which guides display within the "Guides" area on a user's Salesforce page by entering criteria in the guide properties. Enter the criteria in the **Show Guide Only When** field on the Guide Properties **Advanced** tab.

The criteria must be entered as a Boolean expression. You can use the functions and operators described in the Salesforce documentation. See the examples below.

Use the **Insert Field** list to select insert objects and fields into the expression.

This expression can be as complex as you need. For example, here is an expression that tells Guide Designer that it should only display a guide when an account's owner is one of three people:

```
CONTAINS($Account.Owner.Name, "John Smith")
    || CONTAINS($Account.Owner.Name, "Bob Smith")
    || CONTAINS($Account.Owner.Name, "Jim Smith")
```

Note: Do not use Salesforce field syntax (enclosing field names in curly braces with the "!" prefix) when you enter field names. The expression is passed directly to Salesforce as an Apex expression. If you pass an invalid Apex expression, Salesforce returns an error message.

Example 1

If the field is a list, you can use an equals sign to select one of the items from the list. For example:

```
Account.Type = 'Analyst'
```

Example 2

To define multiple criteria, use "&&" (logical AND) and "||" (logical OR) in your expression. For example:

```
(Account.Type = 'Analyst' && Account.testfield__c= true) || (Account.Phone ='1234')
```

In this case, the expression means that:

1. The account type must be Analyst AND the testfield must be true, OR
2. The account phone number must be 1234.

In this case, the parentheses group the first two tests so they are evaluated as a single unit.

In the following example, the parentheses cause this expression to be evaluated differently:

```
(Account.Type = 'Analyst') && (Account.testfield__c= true || Account.Phone ='1234')
```

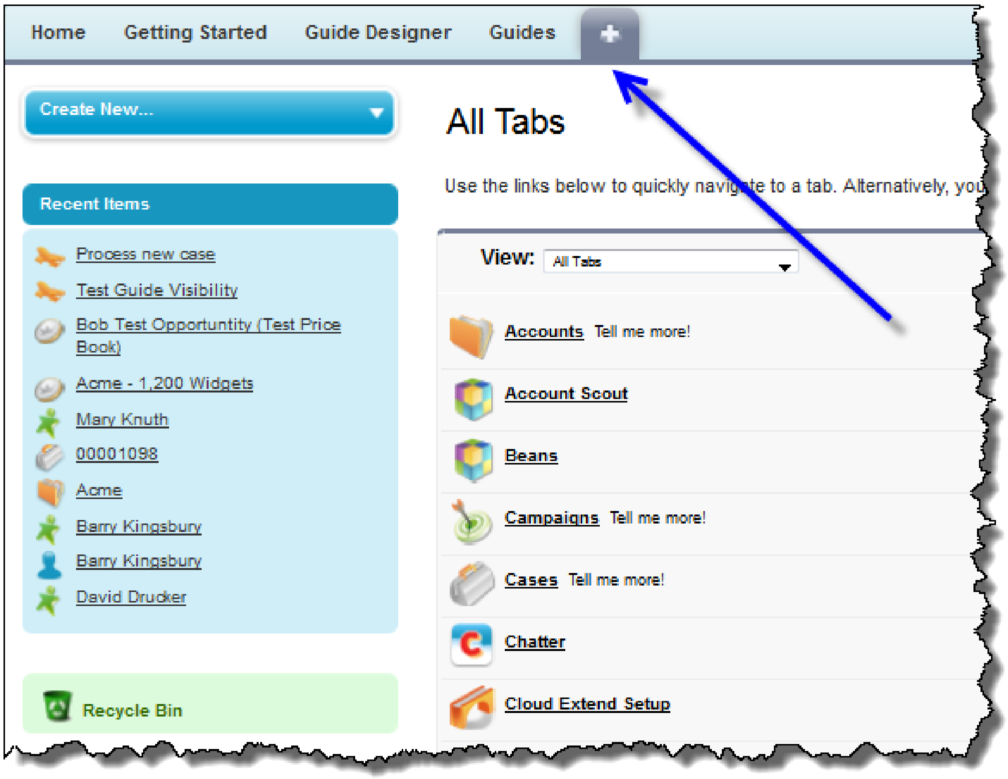
Here, the same criteria means that:

1. The account type must be Analyst AND
2. Either the testfield must be true OR the account phone must be 1234.

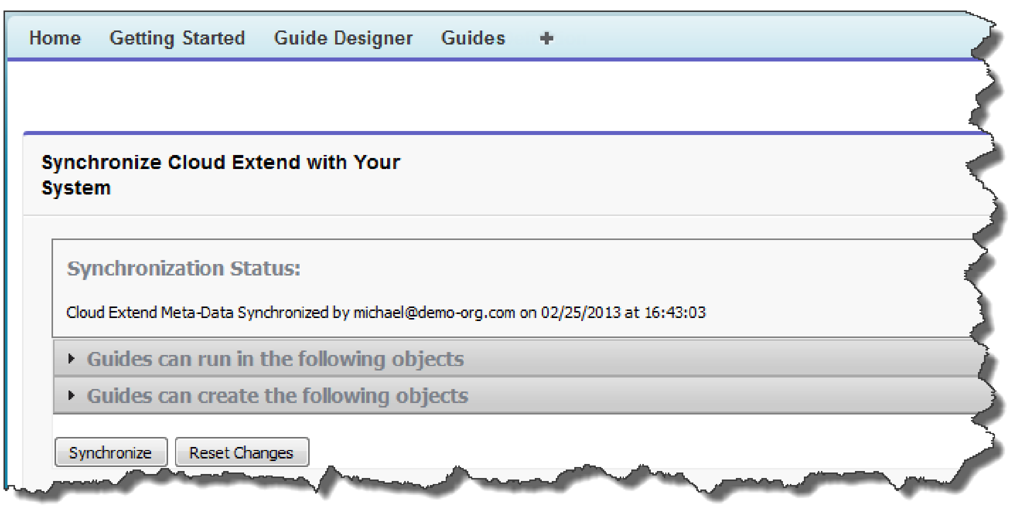
Creating an Object that isn't Available by Default

Guide Designer knows about many of the objects you use in Salesforce. There are others that Guide Designer doesn't automatically know about. In addition, your site may have custom objects. You can add these objects to Guide Designer using the following procedure. In this example, Contact Roles for an Opportunity are being added.

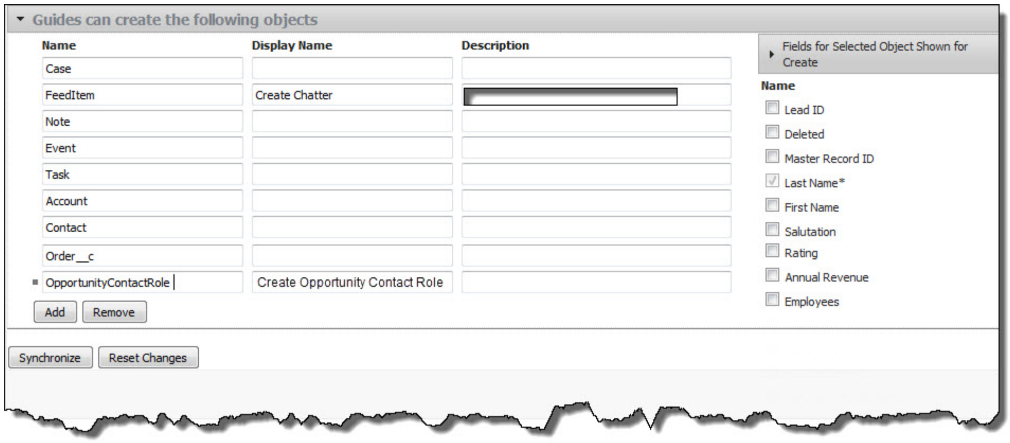
1. Go to Guide Designer Setup by pressing the "+" icon. Here's an example:



2. Click **Setup**.
   Here's what you should see:



3. Expand the *Guides can create the following objects* section and press the **Add** button to create a new object:



As you are typing "OpportunityContactRole", Salesforce displays a list from which you can select objects that are set up in Salesforce. After the name is entered, you will see the object's field on the right. Notice that required fields are checked automatically. If you want other fields to be displayed by default in the Guide Designer when doing a "Create Opportunity Contact Role", you can add them by checking their name.

4. After this object is defined, press the **Synchronize** button.

If after you synchronize, you do not see this object, try refreshing or restarting your browser.

CHAPTER 8

Launching a Guide

This chapter includes the following topics:

Launching a Guide Overview

You can launch a guide from Application Integration or from Salesforce.

If you want to launch a guide from Salesforce, you must perform the following prerequisite tasks in Application Integration:

1. Create a guide and select a Salesforce object in the **Start** > **Applies To** property. The guide gets associated with the object that you select.
2. Validate, save, and publish the guide.

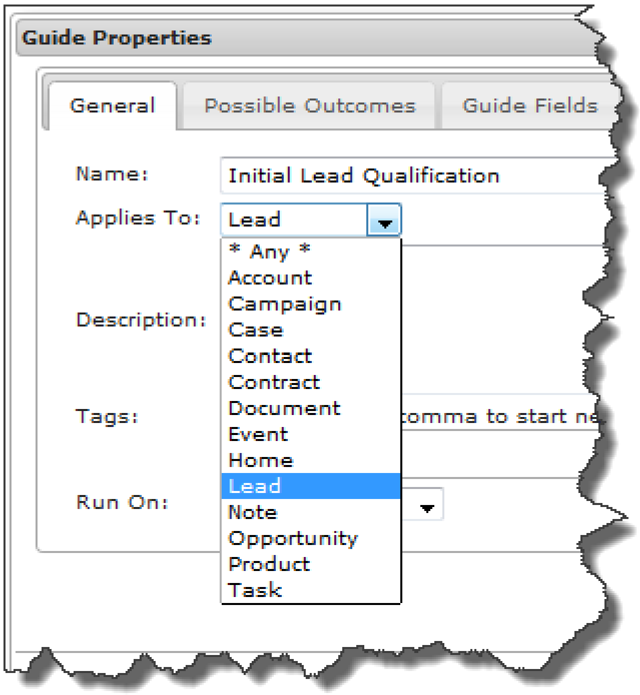
Note: If you log in to Salesforce and do not see your guides, log out of Salesforce, clear the browser cache, and then log in to Salesforce.

You can launch the guide from the Salesforce home page or from a Salesforce button. You can also create a calendar event or a vCard in the Guide Designer, to launch a guide from your smart phone.

Launching Guides from a Home Page

When users log into Salesforce, they are logging into their Home page. This is a natural place from which they can launch many guides. Any guide, however, that is launched from a Home page does not have a context; that is, the guide does not apply to any object. (If it applied to the object, it would appear on the object's page.) Consequently, home page guides are often used to create Salesforce objects.

To place a guide on the user's home page, set the **Applies To** field within the **Guide Properties** dialog to **Home**.



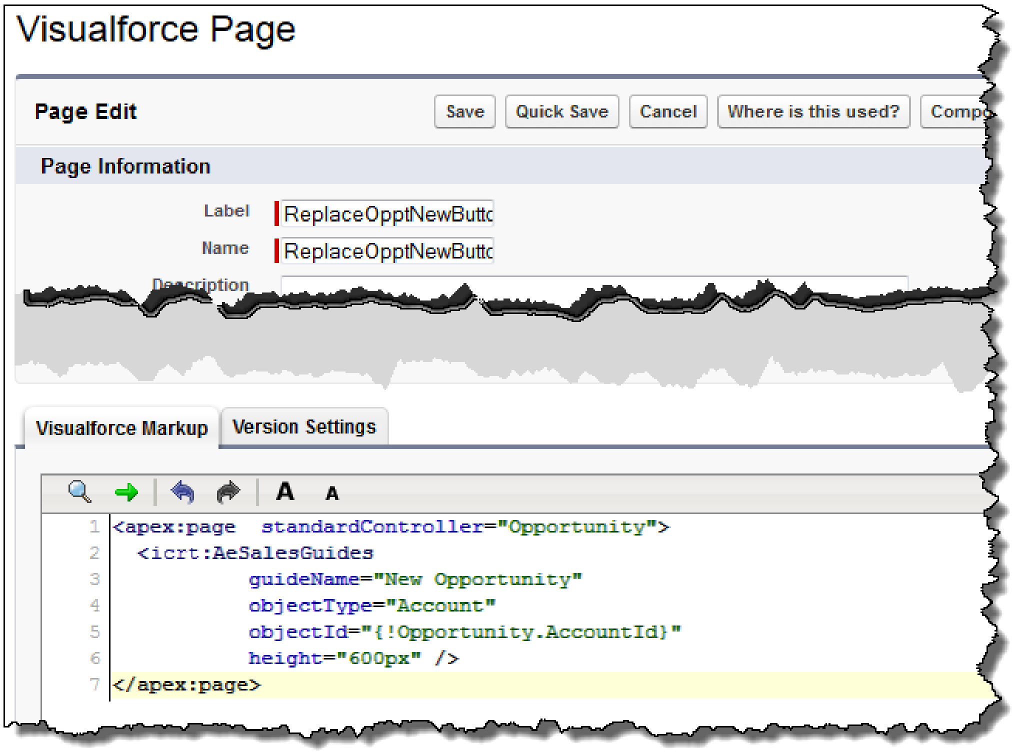
Launching a Guide from a Salesforce Button

The previous topics in this section showed how you can add a button to a page and have that button launch a guide. In some cases, you want a guide to launch when a user presses a button. For example, within an Account page, pressing the "New Opportunity" button tells Salesforce to display an empty Opportunity page. This topic shows how you could instead launch a guide when the user presses the button.

Step 1: Create a Visual Force Page

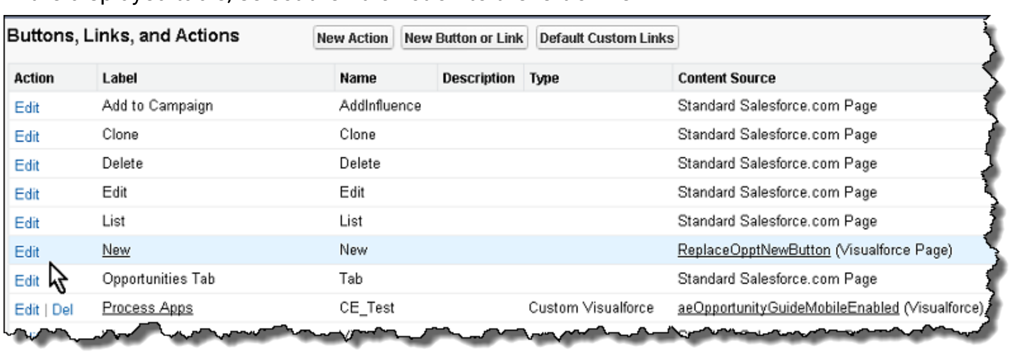
1. Select **Setup > App Setup > Develop > Pages**. At the top of the table that is displayed, press the **New** button.
2. Enter the label that will be used for your code. Salesforce will also automatically give it a name. You can give it another name if you like.
3. Add your code to this form. The following code is designed to modify the Salesforce built-in Opportunity object. It will invoke a guide called "New Opportunity". As it will be invoked from within an Account page, the `objectType` is set to `Account`. The opportunity will need to know the `AccountId` (which is the same as when the form is displayed when pressing the **New** button). The final attribute indicates the guide will display in an area whose height is 600 pixels.

```
<apex:page standardController="Opportunity">
    <icrt:AeSalesGuides
           guideName="New Opportunity"
           objectType="Account"
           objectId="{!Opportunity.AccountId}"
           height="600px"/>
</apex:page>
```



Step 2: Reassign the "New" Button

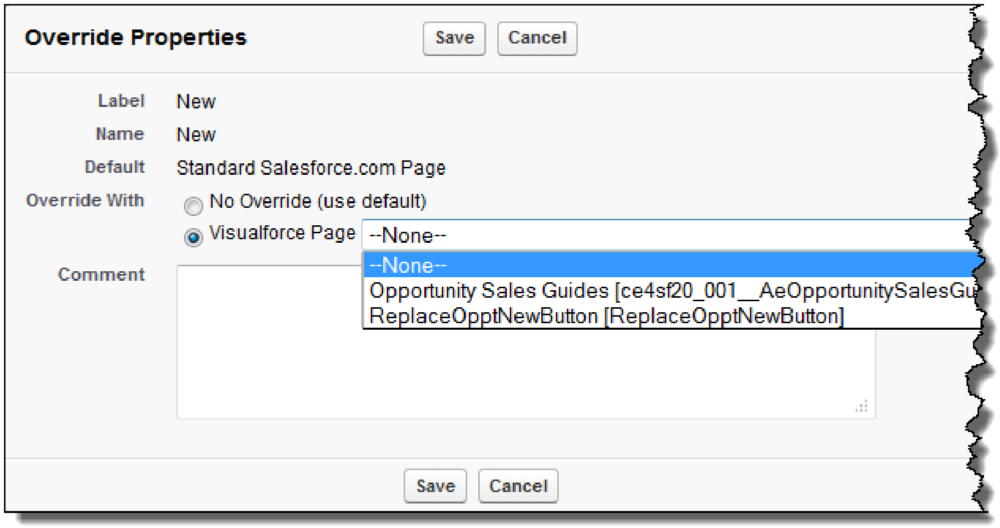
1. Select **Setup > Customize > Opportunities > Buttons, Links, and Actions**.
2. In the displayed table, select the **Edit** Action to the left of New.



Buttons, Links, and Actions — New Action | New Button or Link | Default Custom Links

Action	Label	Name	Description	Type	Content Source
Edit	Add to Campaign	AddInfluence			Standard Salesforce.com Page
Edit	Clone	Clone			Standard Salesforce.com Page
Edit	Delete	Delete			Standard Salesforce.com Page
Edit	Edit	Edit			Standard Salesforce.com Page
Edit	List	List			Standard Salesforce.com Page
Edit	New	New			ReplaceOpptNewButton (Visualforce Page)
Edit	Opportunities Tab	Tab			Standard Salesforce.com Page
Edit | Del	Process Apps	CE_Test		Custom Visualforce	aeOpportunityGuideMobileEnabled (Visualforce)

3. In the displayed Override Properties page, go to the bottom of the form and set Visualforce Page. From the picklist, select the name of the Visualforce page This is the name you set in the previous section.



4. Press the **Save** button.

Step 3: Test that the Guide is Launched

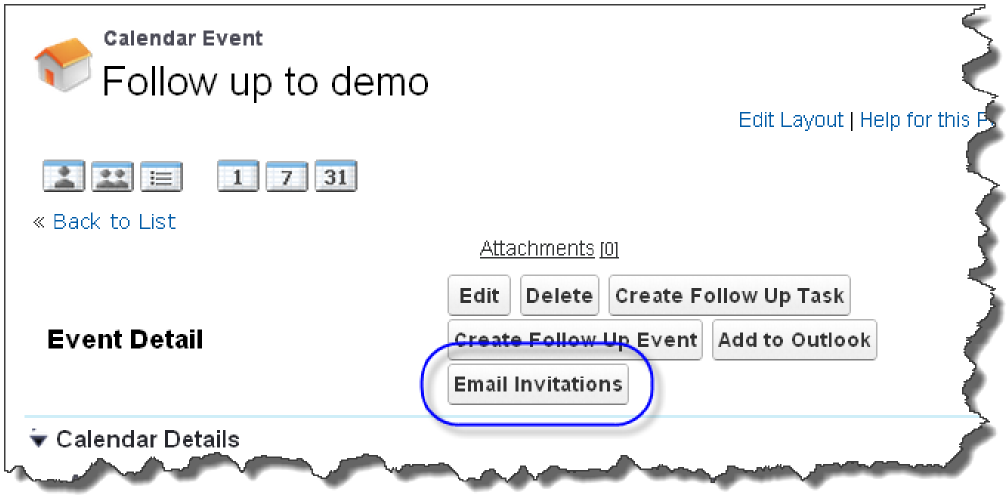
1. Open a page for an existing account.
2. Scroll down the Opportunities area, then press **New Opportunity**. The New Opportunity guide should launch.

Launching Guides from a Calendar Event

Guide Designer lets you to send iCalendar files to people who will be attending an event. When it was installed, Guide Designer added an **Email Invitations** button onto Calendar Event pages that helps you do this.

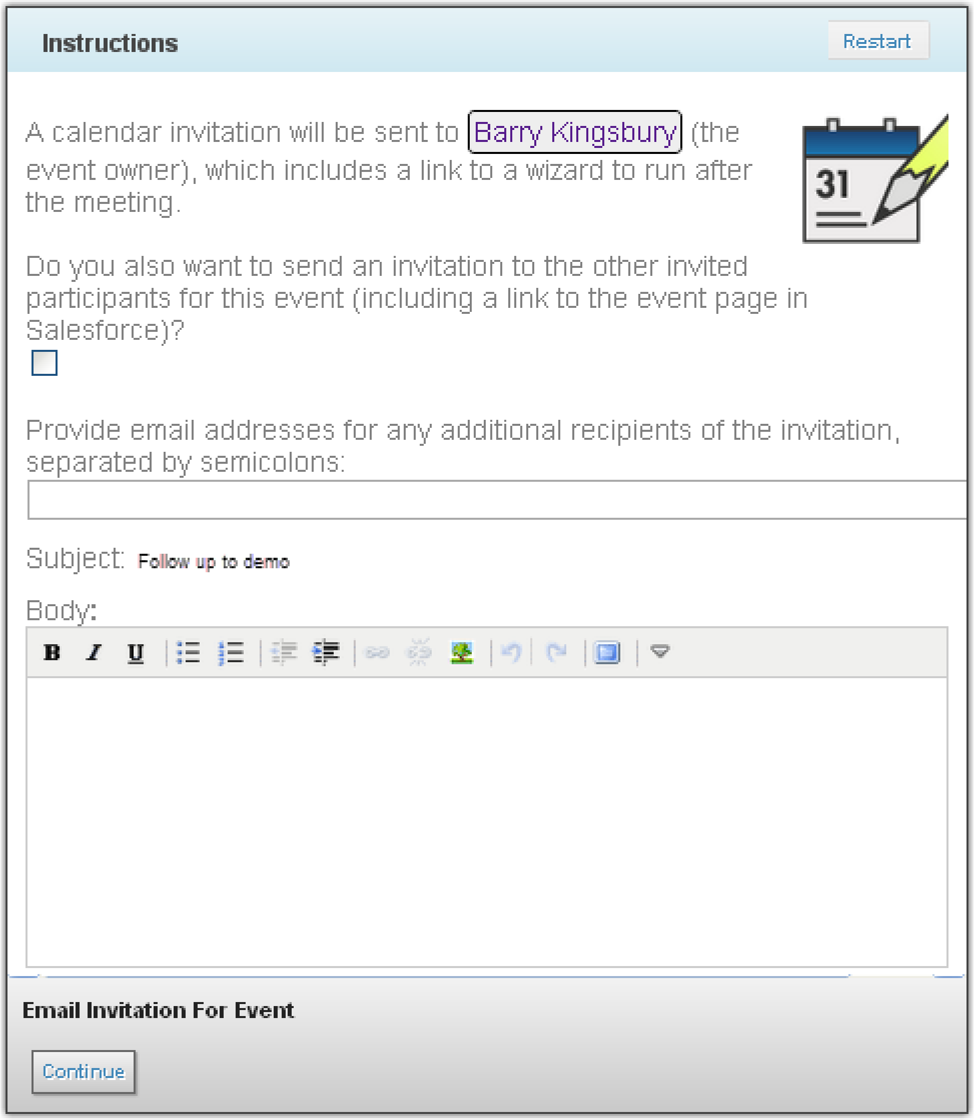
Here's how you send iCalendar files to meeting participants.

1. Create an event in Salesforce. Generally, creating an event is separate from sending invitations, although it doesn't have to be. As you usually do when creating an event, you will identify the person the event is assigned to, the object such as an opportunity that the event is related to, and so on.
2. After creating the event, go to the created event's page. Here you will see an **Email Invitations** button.

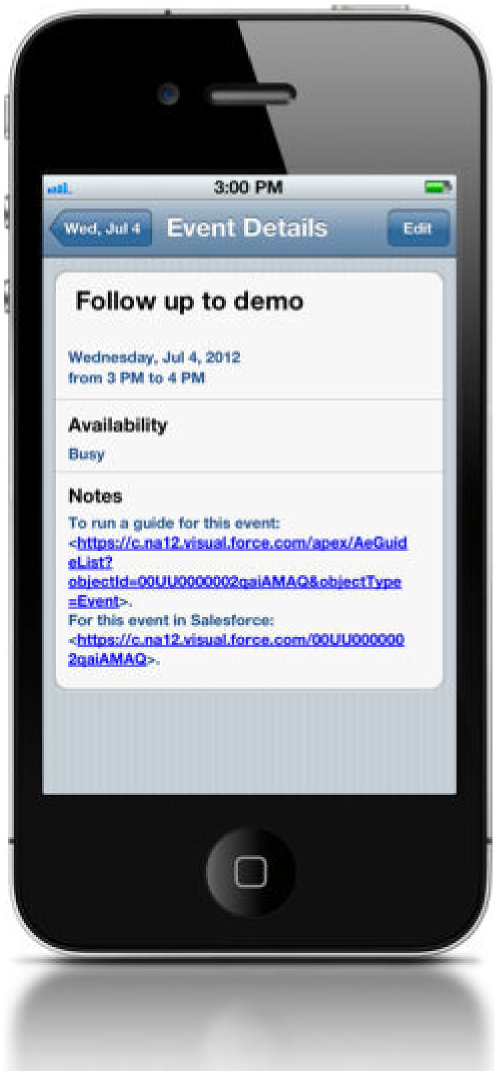


3. Press this button to display a wizard.

image href="../images/EmailInvitations.png"



4. After filling out this page, click the **Continue** button. Guide Designer will now create an iCalendar file for the event, attaching it to an email that it sends to people participating in the event.
   Note: Guide Designer can create two different iCalendar files. The first is included in emails sent to people who are named as contacts, leads, and the like. This iCalendar file lets them add the event to their calendar. In contrast, the person sending the invitation receives an email that contains a link to the event. Also, if the invitees include other users who are working for your company, they also get a link to the Event object. This lets them get more information about the event. However, only the event's host (the one to which the event is assigned) can run the event.
5. Open the calendar attachment on your phone. (If you open it on a PC, you will need to sync it to your phone). Here is the kind of information that the person who is hosting the meeting sees on an iPhone.



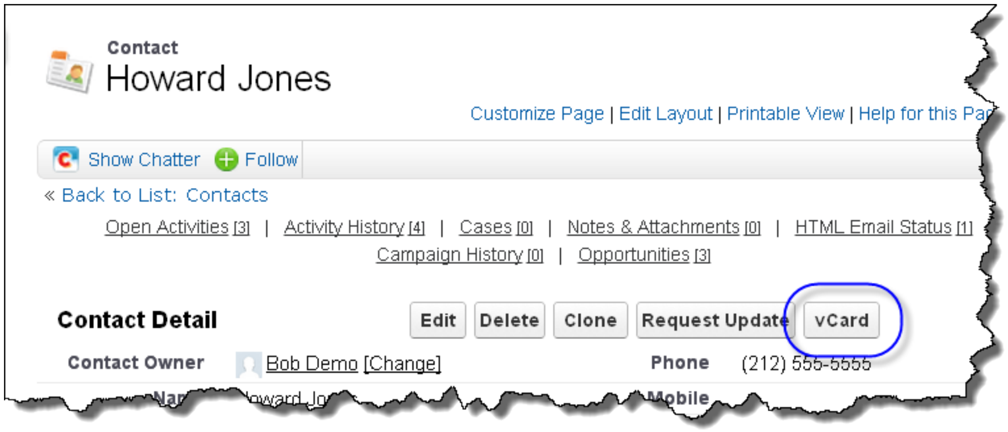
6. At the time of the meeting, users can tap the link to access the guide.

Launching Guides from a Phone Contact

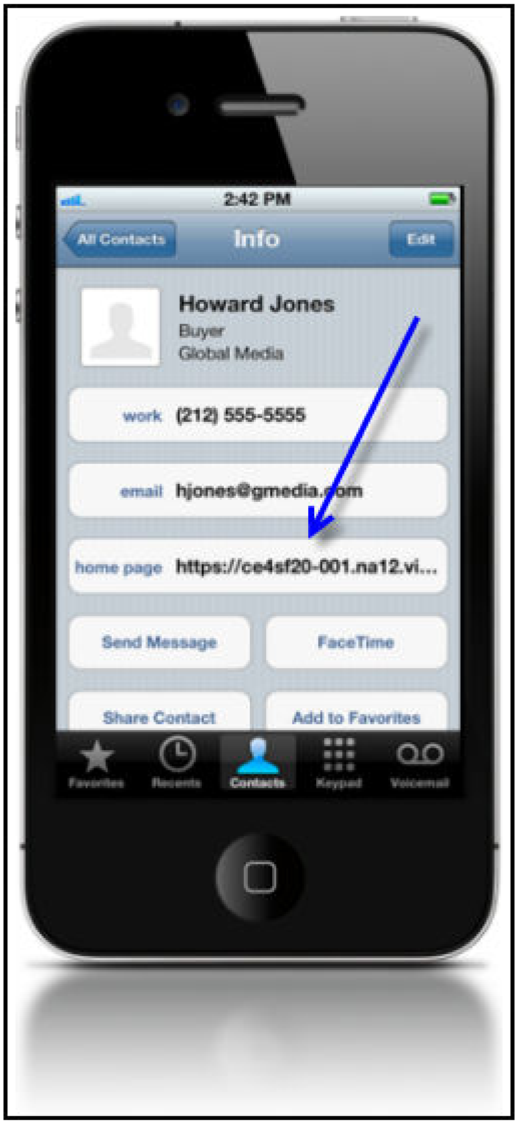
When Guide Designer is installed, it added a **vCard** button onto Contact pages. After pushing this button, Guide Designer creates a `vCard` file that you can use to add this person to a contact list. When you are in your contact manager, the information displayed when you open the contact contains a link that when tapped displays the guides that a user can run for this contact.

Here how you do this:

1. Click the **vCard** button on a Contact page.



2. After you press this button, your browser receives a **vCard** file. Execute this file to have it be installed in Outlook or another contact manager. If you do this on a PC or Mac, your contact manager must sync with the contact information on your smart phone. Here is what a contact page looks like on an iPhone:



3. Notice the link in the home page area. Tapping this link displays the guides that you can run for this contact. If there is only one guide, Guide Designer displays that guide.

CHAPTER 9

Guide Execution Reports

This chapter includes the following topics:

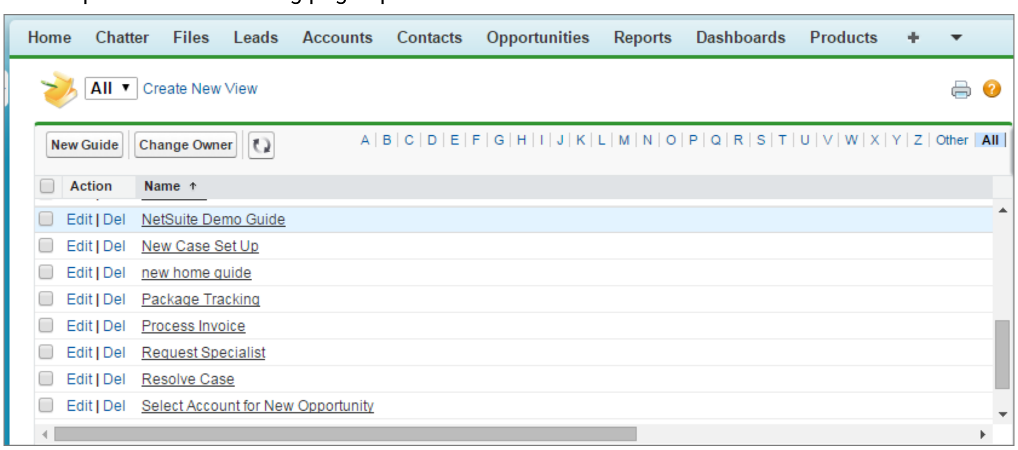
Guide Execution Reports Overview

This chapter describes how to view guide execution history from the Guides tab and use the Salesforce Reports feature to run a report on Guide Executions.

Using the Guides Tab

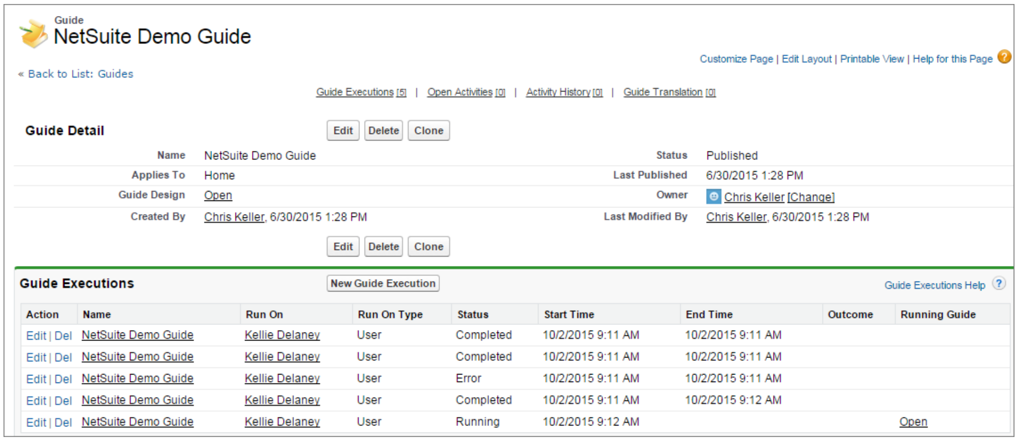
To view guide execution history from the Guides tab:

1. Locate the Guides tab in the Salesforce menu. You may need to click the **+** sign and choose it from the list of options. The following page opens:



Note: The Salesforce controls on this page apply only to the list. You must open Guide Designer to use the guide design features.

2. Click on the guide you want to view to see the execution details, as shown in this image:

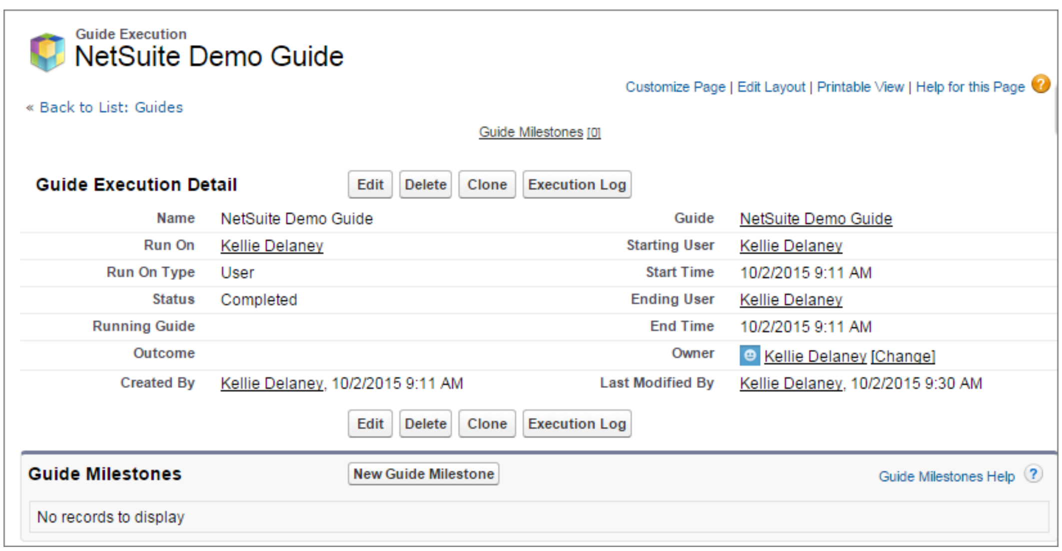


From this page, you can:

- Click **Guide Design** in the header to open the guide in Guide Designer.
- Click the guide name in the list to display a Salesforce detail page for the selected guide.
- Click the name in the **Run On** column to see information about the user account that run the guide.
- Click **Open** in the Running Guide column if the guide is still executing and view the guide in progress.

Guide Details

If you click Name in the Guide Executions list, the following details page displays:



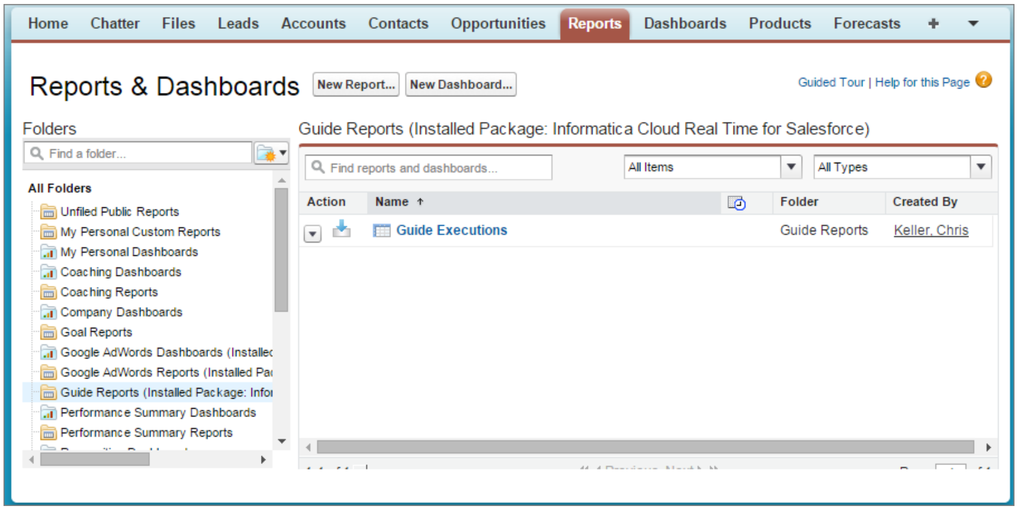
Several options are available to manage the execution reporting history. For example, you can:

- Click **Delete** if you no longer need this history. However, note that if you delete the guide history from this page, you must republish the guide in Guide Designer before it is available for execution.
- If the guide includes milestones, click **Milestone** to display a detail page for the milestone.
- Click **Execution Log** to log in to Informatica Cloud® and review the guide execution log in the Application Integration Console.

Creating Salesforce Reports

To use the Salesforce Reports feature to run a report on Guide Executions:

1. Choose the **Reports** tab in Salesforce (or click the + sign to locate it).
2. Choose **Guide Reports** from the list. The following appears:



3. Click **Guide Executions** to run the report and view options to change report parameters. Use the Salesforce commands on this page in the same way you would use them on other report pages.